Heuristic-1

Inspiration: Aggressive play by punishing the opponent more than rewarding myself.

$$my_{moves} - \alpha\,(opponent_{moves})$$

A value of $\alpha = 2.0$ is typically suggested, but a value of 2.5 instead seems to perform slightly better. This may just be an artifact of the game initialization itself. However, use of genetic algorithms can provide a good number for α. I have used $\alpha = 2.0$ for this heuristic.

Heuristic-2

Inspiration: Aggressive play by highly rewarding myself if my_moves is greater than the opponent_moves. Compared to heuristic-1, this heuristic rewards more towards the end game and when the difference in remaining legal moves for both players is less.

For example, if my_moves = 49, and opponent_moves = 24, heuristic-1 will give a value of 1. However, heuristic-2 will produce a value of 1.021. This difference in heuristic values increases more towards the end game. If my_moves = 5, and opponent_moves = 2, heuristic-1 will again give a value of 1. However, heuristic-2 will produce a value of 1.25. Thereby, heuristic-2 may perform better than heuristic-1, by inherently taking in account the game state.

$$\frac{my_{moves}}{opponent_{moves}}$$

Heuristic-3

Inspiration: Change the game style according to the game state. A complex heuristic may be computationally expensive, therefore the idea is to use a simple heuristic at the beginning of the game, and switch to more complex heuristics as the game progresses and as every move becomes increasing more crucial.

Game state is given by the ratio of blank spaces to total spaces on the board.

$$game_{state} * (\,my_{moves} - \alpha\,(opponent_{moves}))$$

$$+$$

$$(1\text{-}\ game_{state}) * ply\_ahead\_my_{moves} - ply\_ahead\_opponent_{moves}$$

I have used $\alpha = 2.5$ for this heuristic.

Heuristics comparison

Results of the heuristics are shown below. As I expected the AB_custom_3 player outperforms other players by significant margin, included the provided AB_Improved player. The reason is that AB_custom_3 is looking one move ahead in the tree, hence making more “intelligent” best move decisions.

Match #	Opponent	AB_Improved		AB_Custom		AB_Custom_2		AB_Custom_3	
		Won	Lost	Won	Lost	Won	Lost	Won	Lost
1	Random	48	2	42	8	43	7	48	2
2	MM_Open	25	25	31	19	30	20	35	15
3	MM_Center	35	15	35	15	37	13	44	6
4	MM_Improved	21	29	27	23	29	21	37	13
5	AB_Open	27	23	22	28	24	26	29	21
6	AB_Center	21	29	28	22	29	21	34	16
7	AB_Improved	23	27	32	18	28	22	37	13
	Win Rate:	57.1%		62.0%		62.9%		75.4%	

Recommendations

I recommend using the heuristic-3, because of the following reasons:

1. AB_Custom_3 performs much better against MM_Improved compared to other heuristics. AB_Custom_3 wins 74% times against MM_Improved compared to the next best of 58% by AB_Custom_2 against MM_Improved. Similarly, AB_Custom_3 performs significantly better than other players against AB_Center and AB_Improved players, which I found were difficult to beat with a high win rate.
2. AB_Custom_3 mimics "intelligent" play by inherently getting more thorough about the next move as the game progresses, thus reducing a chance of making a bad move in the crucial stages.
3. AB_Custom_3 performs at a win rate of 75.4% overall against all the opponents combined, which is almost 12% higher than the next best player.
4. Heuristic-3 is an expensive function to compute, compared to other heuristics. But, it makes better move choices as we can see from the high win rate, overall and against tough opponents. Therefore, there is a tradeoff between computational expense and the accuracy. But, the data suggests that it is worth spending some computational time on the heuristic, as overall the win rates do increase. Basically, its worth doing complex calculations (computationally expensive) as long as the best move is close to perfect.
5. AB_Custom_3 uses $\alpha = 2.5$, making it very aggressive right from the initial game state. As discussed earlier, $\alpha = 2.5$ performs better than $\alpha = 2.0$ in the limited number of experiments I conducted.