Heuristics

opp_open_move_score

This is the opposite of the number of moves available to the player opponent plus 8.
So that if the opponent have no moves available it will be 8.
And if it can play anywhere it is 0.

And +inf/-inf in case of terminal states.

free_space

Return the number of free space next to the player.

And +inf/-inf in case of terminal states.

improved_score_with_distance_factor

This is the improved_score from the course but with an added distance factor.
This factor is 0 if the player is at the center of the board and 1 if it is in a corner, and values in between proportionally to the distance from the center.

And +inf/-inf in case of terminal states.

Performances

To evaluate the performances of the chosen heuristics, we first evaluated the performances of a reference agent called ID_Improved.

This ID_Improved agent implement alphabeta pruning with iterative deepening and uses the improved_score heuristic described in the course.

We measure the number of games this agent wins against the following agent profiles:

- Random : random player
- MM_Null : non iterative minimax with search depth of 3 and null_score heuristic
- MM_Open : non iterative minimax with search depth of 3 and open_score heuristic
- MM_Improved : non iterative minimax with search depth of 3 and improved_score heuristic
- AB_Null : non iterative alphabeta with search depth of 5 and null_score heuristic
- AB_Open : non iterative alphabeta with search depth of 5 and open_score heuristic
- AB_Improved : non iterative alphabeta with search depth of 5 and improved_score heuristic

The null_score heuristics returns +inf if the player wins, -inf is it loses and else 0.

The open_score heuristics returns the number of available moves of the active player or +inf/-inf if it wins or loses respectively.
The improved_score returns the difference between the available number of moves of the active player and the available number of moves of its opponent. +inf/-inf if it wins or loses respectively.

The ID_Improved agent performance represent a baseline against which we'll compare the performance of our custom heuristics.

Our custom heuristics performances are evaluated using the same agent configuration than ID_Improved : alphabeta pruning with iterative deepening.

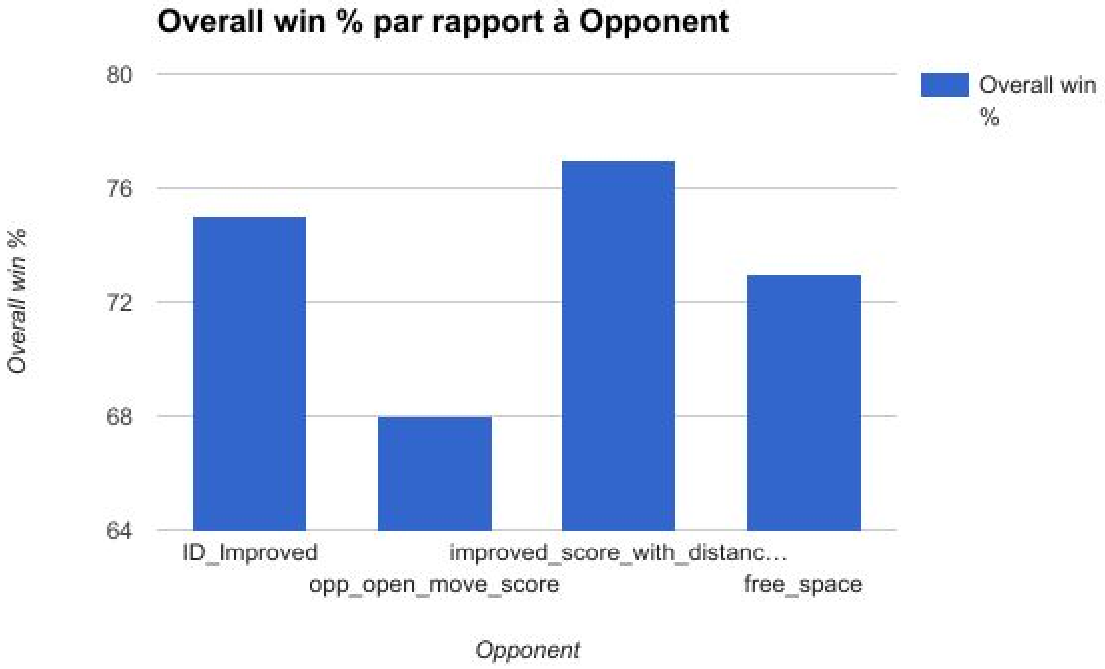
As shown in table 1 and illustrated in graph 1 :

- the opp_open_move_score heuristic underperforms the reference agent.
- free_space agent performance is similar to the reference agent.
- improved_score_with_distance_factor is slightly overperforming the reference agent.

Table 1. Number of wins over 20 games

Opponent	ID_Improved	opp_open_move_score	improved_score_with_distance_factor	free_space
Random	19	18	20	20
MM_Null	20	15	14	16
MM_Open	12	11	16	16
MM_Improved	11	11	15	14
AB_Null	17	15	16	12
AB_Open	14	13	14	13
AB_Improved	12	14	13	12
Overall win %	75	68	77	73

Graph 1. Overall win%



Recommendation

improved_score_with_distance_factor is the heuristic performs the best and should be used.

- overall it performs 2% better than the ID_Improved agent.
- its computation cost is just 5 float operations and 6 int operations more that the ID_Improved heuristic.
- it derives from a simple intuition that a human player would use to play the game.