AIND - Advanced Gameplay: Isolation Heuristic Analysis

Brian Diesel

In order to assist the Isolation game play after implementing minimax search with alpha-beta pruning and iterative deepening I developed 4 heuristics as part of this project. These heuristic functions belong to two basic strategies for game play. One category of heuristic is to use a weighted scaler to the opponent in order to make the more player aggressive. The second category of heuristics involve move based strategies. After playing a few games on paper I decided a good approach was to aggressively play the center in the opening moves of the game which would limit the available future moves of the opponent player. The final heuristic I developed involves a combination of both move scalar heuristics. Throughout the heuristic development I compared my agent to the "Improved" agent playing 100 matches. I found the results from 20 matches had a high degree of variance in the results.

Heuristic 1: This heuristic is very similar to the supplied "improved" heuristic with the addition of a constant scalar applied to the opponent. This scalar is used to amplify the penalty value on moves that do not have a large advantage value. The formula used for this heuristic is:

return number of my moves - number of opponents moves x 2

Heuristic 2: My second heuristic is similar is an iteration of the first heuristic. This heuristic uses a decaying weight value which plays more aggressive in earlier moves and more conservative as the game progresses.

decay = number of blank spaces / area of game board * scalar

return number of my moves - number of opponents moves x decay

Heuristic 3: After experimenting with the weighted strategies without great success I then chose to pursue a heuristic based on positional strategy. Here I used distance dominate the center of the board. From playing initial practice matches it seems that plays in the center seem to result in better success. Here I used distance from the center of the board as a scale factor applied to

move distance from center = square root((move_x_pos - x_center)^2 + (move_y_pos - y_center)^2)

return number of my moves * distance from center - number of opponents moves * distance from center

Heuristic 4: This heuristic assumes the center moves are more valuable early in the game but less valuable as the game progresses. The rational is the center spaces are more valuable because they allow for more freedom in future moves so taking these places early is ideal. This heuristic combines the strategy of taking the center from Heuristic 3 and uses the decaying value from strategy 2

decay = number of blank spaces / area of game board * scalar

move distance from center = square root((move_x_pos - x_center)^2 + (move_y_pos - y_center)^2)

return number of my moves * dist from center - number of opponents moves * dist from center * decay

Percentage of wins in 100 matches

	ID_Improved	Weighted	Decay	Center	Center-Decay
Random	89%	81%	87%	84%	87%
MM_Null	80%	80%	77%	77%	82%
MM_Open	63%	70%	68%	66%	75%
MM_Improved	56%	66%	62%	64%	62%
AB_Null	68%	72%	72%	74%	73%
AB_Open	70%	61%	62%	57%	68%
AB_Average	56%	64%	59%	67%	59%
Final	**68.86%**	**70.57%**	**69.57%**	**69.86%**	**72.29%**

After experimenting with several examples I found some improvement using the heuristic functions. I was however hoping for a more significant final result. I believe a better future heuristic may be to start move the center pieces early, maybe even using an opening book of moves, then perhaps chasing the opponent to block their available future moves.