Heuristic analysis

Jaco Fourie

July 20, 2017

1 Performance evaluation of scoring heuristics

Several different heuristics were designed and implemented. In general I noticed that the simple heuristics that were computationally cheap performed better than those that better estimated the true value of a position at the expense of computational simplicity. My theory is that once a heuristic becomes so expensive that it cannot afford to search deep into the game tree the performance gained by better estimating the value of a move is negated. The aim then is to find a heuristic that is simple to calculate and captures the game logic well enough to estimate the true game state.

I also noticed that there can be a lot of variability in the outcome of a game and that many games need to be played before a decision can be made that one heuristic is better than another. Therefore I increased the games-per-tournament from 15 to 40. To make testing and evaluation of heuristics faster I also chose to only focus on opponents that use full alpha-beta pruning since these opponents almost always perform better than those using only simple Minimax.

1.1 Heuristic 1: Aggressive + corner penalty

This simple heuristic is based on the *improved_score* heuristic that calculated the difference between the players available moves and the opponent's. I noticed that when this heuristic fails to win it is often due to getting stuck in a corner and having the opponent block it's last remaining move with the opponent's next move. The main idea of this heuristic is to penalise a move when it is in one of the 4 corners. I also weigh the number of moves available to the opponent more heavily than those available to the player making this strategy more aggressive and improving the performance. Heuristic 1 is defined as

$$H_1 = \text{Moves}_{player} - 2 \times \text{Moves}_{opponent} - P_{corner}$$

where Moves_{player} is the number of moves available to the player and $\text{Moves}_{opponent}$ is the number of moves available to the opponent. P_{corner} is defined as

$$P_{corner} = \begin{cases} 0 & \text{not a corner} \\ 3 & \text{in a corner} \end{cases}$$

1.2 Heuristic 2: Keep your distance

Heuristic 2 is based on Heuristic 1 but instead of adding a corner penalty I try to prevent too aggressive behaviour by adding a distance between players metric. As the player moves closer to the opponent a penalty is applied to devalue that move. We use the sum-of-absolute-difference (SAD) distance (also known as the Manhattan distance) instead of Euclidean to minimise computational cost. Heuristic 2 is defined as

$$H_2 = \text{Moves}_{player} - 2 \times \text{Moves}_{opponent} + P_{distance} \times \gamma$$

where Moves_{player} is the number of moves available to the player and $\text{Moves}_{opponent}$ is the number of moves available to the opponent. $P_{distance}$ is defined as

$$P_{distance} = \|\text{Pos}_{player}[0] - \text{Pos}_{opponent}[0]\| + \|\text{Pos}_{player}[1] - \text{Pos}_{opponent}[1]\|$$

where Pos_{player} is the position of the player and $\text{Pos}_{opponent}$ is the position of the opponent. The scaling factor γ is chosen experimentally and I found the best results were achieved at $\gamma = 4$

1.3 Heuristic 3: Adaptive combination

Similar to Heuristic 1 this heuristic is based on an aggressive version of the *improved_score* heuristic and adds a penalty when the player moves towards corners. However, I noticed that Heuristic 1 would still get stuck on sides and so the general penalty should be something similar to the *center_score* Heuristic. As with Heuristic 2 the weight of the penalty term was determined experimentally. Heuristic 3 is defined as

$$H_3 = \text{Moves}_{player} - 2 \times \text{Moves}_{opponent} - P_{center} \times \gamma$$

where Moves_{player} is the number of moves available to the player and $\text{Moves}_{opponent}$ is the number of moves available to the opponent. P_{center} is defined as

$$P_{center} = \|\text{Pos}_{player}[0] - \frac{\text{width}}{2}\| + \|\text{Pos}_{player}[1] - \frac{\text{height}}{2}\|$$

and the scaling factor γ is chosen experimentally at $\gamma = 0.5$.

1.4 Results

The performance of the three heuristics as compared to the *improved_score* heuristic is summarised in the table below. With the increased number of simulation games and focusing only on the full alpha-beta pruning agents it becomes clear that none of my heuristics fared consistently better than the *improved_score* heuristic. The best results came from Heuristic 3 which did slightly better than *improved_score* and also happens to be the most similar to it. Heuristic 3 should really be considered a fine tuning of *improved_score* instead of a completely new one.

More complicated heuristics were considered that tried to estimate how many further moves a player could make from a certain position. I quickly realised that these methods are both too computationally expensive and they rely too much on assumptions regarding how the opponent will move. The best heuristics seemed to be the ones that were simple to calculate and captured the flexibility of movement for both the player and the opponent in some way.

Match	**Opponent**	$\alpha\beta$ **Improved**		$\alpha\beta$ $\mathbf{H}_1$		$\alpha\beta$ $\mathbf{H}_2$		$\alpha\beta$ $\mathbf{H}_3$	
		Won	Lost	Won	Lost	Won	Lost	Won	Lost
1	$\alpha\beta$ Open	36	44	41	39	41	39	43	37
2	$\alpha\beta$ Center	51	29	41	39	42	38	47	33
3	$\alpha\beta$ Improved	40	40	44	36	42	38	42	38
	Win rate	52.9%		52.5%		52.1%		55.0%	