AI Essentials Project

I have chosen to develop an AI to play Pentago which uses the minimax algorithm and then applies a heuristic function after a certain number of plies to evaluate how good a game state is for the maximising player.

Minimax is a recursive algorithm that, given a board, considers every legal move and then, for each move, recalls itself with the state of the board after this move has been played as an argument. For games with low computational complexity, minimax recalls itself until it reaches the end of the game tree and 'ripples' values corresponding to winning, drawing and losing back to the root node, alternately passing up the largest and smallest values of the child nodes. The effect of this is that the function ultimately returns the move which leads to the best worst possible end state. However, for more complex games, it is infeasible to search to the end of the game tree in a reasonable time and so it is instead necessary to limit the algorithm to thinking a certain number of moves ahead after which it must apply a heuristic function to evaluate how good a given game state is for it. Despite appearing fairly simple, Pentago is one such game. In fact, a conservative estimate of the average branching factor of Pentago is given by:

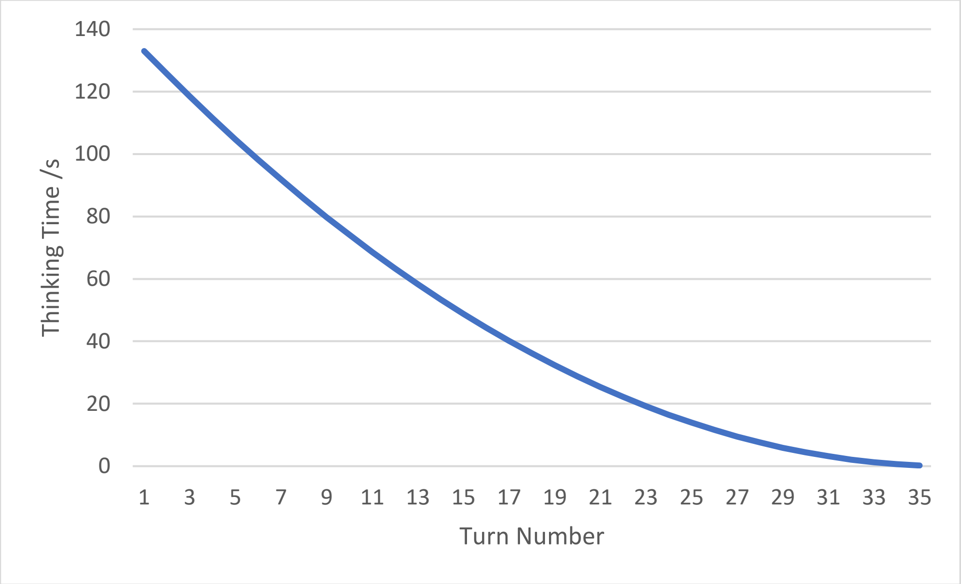
$$\frac{\sum_{r=1}^{36}(296-8r)}{36}$$

The 36s in this expression come from the fact that there are 36 places on a Pentago board and, as such, the longest possible game lasts 36 turns. $296-8r$ is the number of legal moves on turn r. This expression evaluates to 148, although, the actual value of Pentago's average branching factor is higher as most games will not last 36 moves. Nevertheless, this is still a significant branching factor, especially when compared to that of chess which has been calculated to be roughly 35.

Having shown that, for Pentago, the minimax algorithm must be limited to a certain depth, it is necessary to decide on a heuristic function with which to evaluate the 'goodness' of a given board. Since the objective of Pentago is to get a sequence of five tokens, an obvious place to start would be to count the number of sequences of each length that each player and, by assigning values to each length of sequence, calculate a score for each player. The other player's score can then be subtracted from the AI's score in order to calculate how good that board is for the AI. Of course, the problem with this method is that it will reward building a sequence of four more highly than a move which enables a sequence of five to be built if that winning move occurs beyond the depth of the algorithm. In addition, some sequences which can be built in Pentago are inextensible and, as such, are ultimately worthless in the game but valued by the evaluation function. However, this horizon effect is largely unavoidable and can most effectively be minimised by increasing the search depth.

The question that remains is that of what scores should be assigned to different length sequences. Sequences of length greater than or equal to five must be assigned a score that is not reachable through a collection of smaller sequences since winning is obviously always preferable to having many smaller sequences. As such, sequences of five or six tokens are assigned a value of positive infinity. Shorter sequences are assigned values equal to a^3, where a is the length of the sequence in order to ensure that, in general, longer sequences are scored more highly than many shorter sequences.

Due to the computational complexity of Pentago, two was the maximum depth for which the algorithm could return a move in a reasonable time frame on an Intel Core i7-8550U @ 1.80GHz. At this depth, it takes 133 seconds to calculate the first move, although, as shown by the graph below, the computational time decreases as the game progresses.



In terms of its performance in games, the time taken by the AI for each move meant that obtaining a large sample size was impractical. However, out of the ten games it played (5 red, 5 blue) against an AI moving at random, the minimax AI won all ten. In addition, at the time of writing, the AI has won all of its seven games against humans. Although none of its opponents were experienced Pentago players, they all had good special reasoning skills and, in fact, one of them played chess at a high level so this statistic is encouraging.