CS B551: Elements of Artificial Intelligence

Enrique Areyan

**Homework 2**

*Answers to Written Questions:*

1 - My evaluate function takes into account various pieces of information to evaluate a state. Given a state and both players, the function does the following for each player:

I. First, it calculates the number of openings for the opposite player and then adds:
II. a weighted sum of available pieces, were the weights are a function of the size of the pieces. The smaller the piece, the greater the weight thus, giving priority to saving the bigger pieces for the end. Next, take away:
III. the number of concealed pieces for this player. The idea is that a smaller number of concealed pieces is better for the player.

Finally, the evaluate function is the difference between evaluating the function for each player. If MAX has a greater advantage, it will be reflected as a large positive number, otherwise a negative number will give MIN the advantage.

I believe this is a good measure of a state's favorability to the players because it takes into account three key variables when playing goblet, i.e., (1) number of spaces available to construct a line (the objective of the game), (2) available pieces taking more into consideration larger pieces which can conceal other pieces, and (3) the number of pieces concealed which gives and idea of how many pieces are not immediately available for play. Moreover, the evaluate function adds or subtracts each of this metrics according to whether it help or hinder a player’s chances of winning the game.

2 - Given a default limit on the number of expansions *M*, and assuming a maximum branching factor of 32, minimax will expand a complete tree of depth at most floor($\log_{32}$(*M*)). This is because h = 32^M. More generally, assuming a branching factor *b*, then h = floor($\log_b$(*M*)).

3 - For the alpha-beta search the largest horizon possible without encountering the default limit on expansions is 1. In other words, with a horizon as low as 2 the search encounters the expansion limit as checked in the *terminals_checks*() function by testing state.expansions_count() <= 0. Note that the default limit on expansion is 15. If we set the default limit to 32, then we can have a horizon of 2 without reaching such limit. The number of expansions must grow exponentially on the horizon in order to not reach the limit.

4 - The maximum number of expansion per ply is the key variable when playing the alpha-beta against the minimax player. If an appropriate large number of expansions are allowed, the alpha-beta player always wins regardless of whether it plays first or second. However, bellow a certain threshold of approximately 1025 expansions -for my particular implementation-, the minimax player wins. This occurs because if we allow the alpha-beta to explore more of the three it will reach stages of the game that the minimax can't reach and thus, would have more information to make better moves.