B351 Final Project Report

Gomoku Game

Yixuan Wang yixuwang@iu.edu Yaodong Chen yaodchen@iu.edu

Introduction

Gomoku, also called Gobang or Five in a Row, is an abstract strategy board game. It is traditionally played with Go pieces (black and white stones) on a Go board. The game is known in several countries under different names. Players alternate turns placing a stone of their color on an empty intersection. The winner is the first player to form an unbroken chain of five stones horizontally, vertically, or diagonally. (Wikipedia)

Problem Space

Gomoku is game that favors the first-hand player. If there is no restrictions on the first-hand player, which is the black stone, can always win the game. (Gomoku and Threat-Space Search, L.V.Aliis) Our approach on the Gomoku AI is to have three levels of difficulties so that the player can have an idea how our source code is affecting the AI. We also have an option to let the AI plays the first-hand to make the game more intuitive since the player may have a willing to player the latter-hand.

Algorithm

We tried two algorithms when developing the game. Tree-based search with Min-Max and Enumeration Method. We decided to use Tree-based search on the easy mode and Enumeration Method on the other two modes.

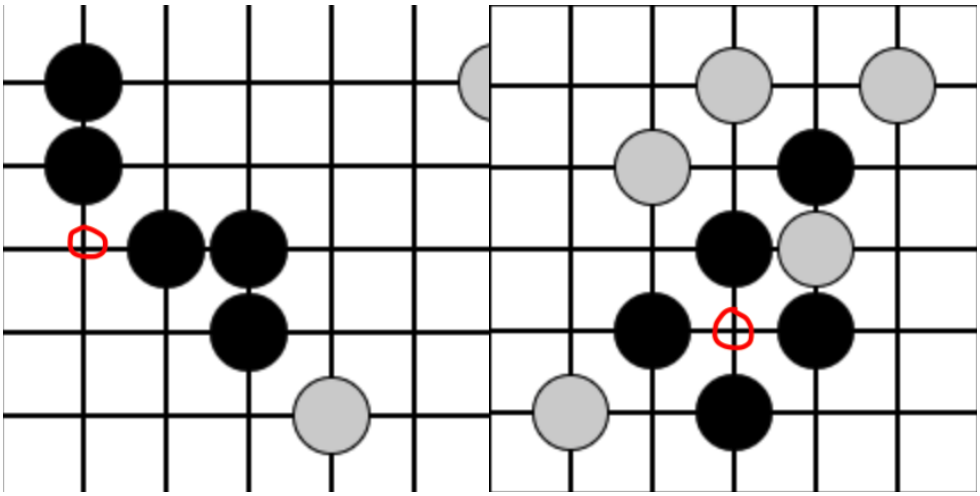
For Tree-based search, the progress of the game is a tree. If the player moves first, then depth 1 is all the possibility the player can move and depth 2 is AI's possibilities. If we assume there is

50 variations for each move and the depth limit is 4, which means the player and the AI move two-hands each, the final depth will have 50^4 = 6.25*10^6 nodes. And a heuristic function is used in each depth to give a evaluation for the current node.

For Enumeration Method, this is the method that we developed ourselves with no reference on the assignments we did in this semester. Because the dimension of the board does not change, we store the board as an 2D array. If the board is 18*18 then there is 896 ways to win in total. The player and the AI each has an array to store these 896 ways to win, when a piece is put in the board, the winning-way that needs this piece will increase and if this piece is opponent's piece, the winning-way will have an abnormal value means this particular winning way is impossible. When the AI decides where to put the piece, an evaluation process will give a score to all the empty intersections on the board and the score can determine in certain scenario, the AI is more incline to intercept the player or connect its own pieces.

Optimization

After some tests on the AI and some research on the Gomoku game, we found that there is particular formations that the black stone can make to force a win more easily. Here are two most common scheme of force a win.



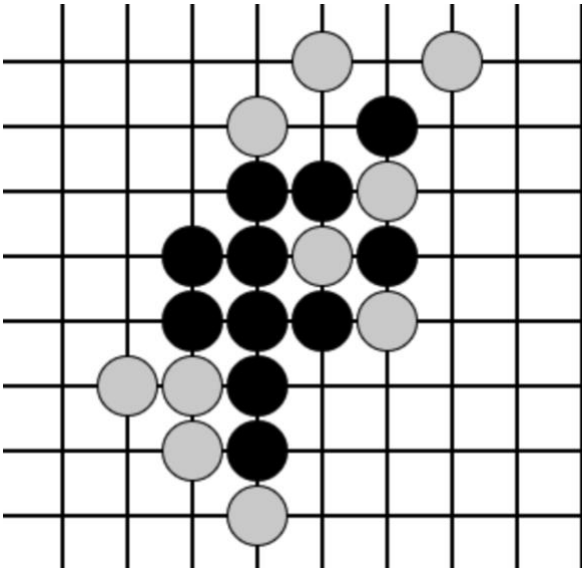
If the black stone has the red intersection, then the black can have a two-way three in a row. Obviously when there is a four in a row, it is a forced win. And two-way three in a row will inevitably cause a four in a row. So we did an optimization on these two scenarios that AI will block the red intersection when there is a formation like the images above.

Tree-based Search VS Enumeration Method

In our project, we have a finite state of environment, which means it is possible to use Enumeration Method to get a best solution for each move. The down side of enumeration also did not appear in our Gomoku game because the computation is well handled by modern computer's. The AI can put down a piece immediately after the player make a decision. During the attempts to use tree-based search algorithm, we found the winning rate of the AI and the decision AI makes does is not really different from the enumeration method. Here we believe enumeration method is the tree-based search with the largest depth limit, so we use it in the hard mode. Enumeration uses the final outcome to predict

Force Win Scenario

Developing the game is also a process of learning how to be better at the game. And we did a lot of research on how to create a force win scenario. Although we keep improving our AI but since the game favors the first hand, the AI will not be able to block the player efficiently if the player figures out a certain track of moves. Here is one example.



Conclusion and Future Work

There are some disadvantages of our AI. Because enumeration method predicts move at an global level, the AI move will often lose local advantages and go for a spot that is better on the entire board. And human player can use that disadvantage to trap the AI. An algorithm that heavily rely on a heuristic function will react exactly the same if the player plays the same hands every time. That makes our AI predictable if the player practice for a few rounds. The solution for that is to add machine learning idea to the AI and AI can improve itself based on the results of each rounds. Our implementation on AI does not learn from each individual game since we did not use any knowledge of machine learning or neural network. We can also have more game elements such as score board and AI plays against each other. The professional Gomoku game actually has many restrictions on the first-hand player, we can also import some rules to our game.