BIphaGomoku

Name: Hao Bai
ID: 180223545
Version: 1.5.0

Algorithm

1. This Gomoku bot is using the minmax search and alpha-beta pruning.

- Search w ill find all available points on the board.
- Firstly, try each available points, and calculate using minmax search and alpha-beta pruning. If the trying location can lead our w in. This time move is this Point.
- The alpha value is the w orst case for maxmizer. The aim of the maxmizer function maxmum our round board value. If the value of the board is less than the alpha value, the search w ill never search the follow ing available points. Which reduced search time.
- The beta value is the w orst case for minimizer. The aim of the minimizer function is to minimize the opponent round board value.If the value of the board is higher than the beta value, the search w ill never search the follow ing available points. Which reduced search time.
- There are 4 layers for each bot to search and calculate the best point for this time to go.
  - Sequence:
    - Maxmizer(bot round)
    - Minimizer(opponent round)
    - Maxmizer(bot round)
    - Minimizer(opponent round)
- After calculate, the algorithm w ill return a list of the candidate points. And calculate the point value of each candidate point. And return the best one.

1. The value function:

- The board value is calculate the number of contiguous point.
  - The blocked contiguous point:
    - 1 point got 1
    - 2 points got 10
    - 3 points got 500
    - 4 points got 10000
    - 5 points got 100000000
  - The unblocked contiguous point:
    - 1 point got 1
    - 2 points got 100
    - 3 points got 9900
    - 4 points got 10099
    - 5 points got 100000000

Experiment:

In experiment, the value function each points value get and the number of layers is modified. It is found that the increasing of the number of layers cause the bot more smart. But the complex layers w ill cause time out even using pruning method. Changing the value geting w ill influence the bot to choose points. For example, if the blocked 3 contiguous point value is set higher than the unblock one. The bot w ill perfer the blocked 3 contiguous point. And finally influence the probability of w inning.

- Testing w ith random: 100% w inning
- Testing play w ith itself: The Nash equilibrium of a good bot need to reach draw . How ever, the 4 layers search sometime could

not reach draw in self play. It might because when gathered point the bot might get higher value in other point not the key points.

- Testing with human: 40% winning. Unfortunately, as a human being, sometimes I can't win.

More information will be posted on this bot github page: BlphaGomoku