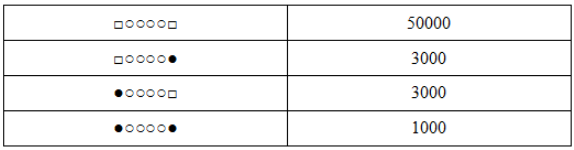
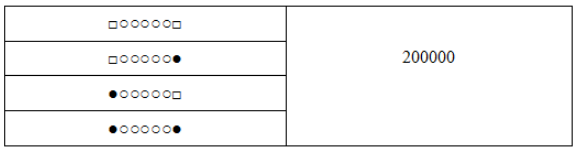
Combinatorial Game - Summary-02

1>Who is the winner

There is no need to search the whole chessboard to judge who is the winner, just test with the latest step.

2> Table for Evaluation

This table is used to calculate the score of the form which we produced by setting the chess. We can get this from the Internet.



3> Evaluate Function

There are two kinds of evaluatee function, one is for the whole chessboard(as what I have talked last time), another is for the single position.

After each step, we should make an evaluation. But there is no need to do this for the whole chessboard because there some blank or the surrounding does not change. In a word, we should just evaluate the position at the eight directions of the latest step.

The Computer should no only consider about itself but also pay some attention to the attacker. It means if the defender can not get the best position to defender it should get the best position to attack, the last choose should choose a “random” position.

Maybe the calculate formulation of the score should be the score the defender get plus the score the attacker get.

4> Get the best position for the computer( as Defender)

This part is the biggest problem, there is no doubt that we should use the tree search algorithm to get the best position. After each step the chessboard will be a root of a new tree then we will produce many branches, obviously, the innovation should be based on how to search the tree quickly and get the optimal position.

The most used method is Alpha-Beta, maybe we can change something about this method to get a better performance. In addition, the metric of the algorithm should be the number of leaf nodes which we have visited.

At the same time, sort the node in some metric will be good for pruning. So I think we should put some attention on the data structure of the game tree.

5> Boyer-Moore Algorithm in Pratten Recognize

Some paper I read said we can use the BM algorithm in such project.

REFERENCE

[1] https://www.cnblogs.com/matrix-r/archive/2013/08/09/3249068.html

[2] <https://blog.csdn.net/lzhuangfei/article/details/80289417>

[3] 张明亮. 一种新的博弈树搜索算法及应用研究[D]. 苏州:苏州大学, 2007.

[4] 张佳佳. 五子棋对战平台的设计与实现[J]. 电脑知识与技术, 2012, 8(22): 5409-5411

[5] 李舒婷. 基于Android的五子棋游戏的设计与实现[J]. 软件导刊, 2016, 15(4): 98-99

[6] 李霞丽. 基于棋型的藏族久棋计算机博弈研究[J]. 智能系统学报, 2018, 13(4): 577-583