Name: Keenan(Kihyun Byun)

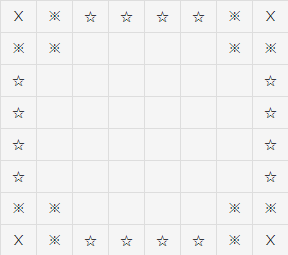
S/N:301381767

Date: Aug 12 2020

CMPT310 Final Project: Reversi

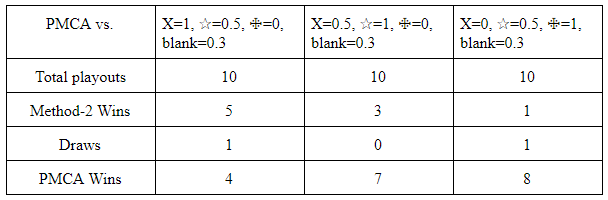
This Reversi-Bot is implemented with the Monte Carlo Tree method. When a user or a program input a current grid to the Reversi Bot, Reversi-Bot randomly simulates many cases (variable ‘N’ times in findeBest function) per each possible movement from the grid. When the simulation is done, Reversi-Bot decides the best movement from its rates of wins: max(number of win/number of plays). This AI included two methods to improve its performance. The first method (method-1) is minimizing the win rate of the opponent player. In the conclusion function in main.cpp file, decision cares next turn; for example, as the AI decides a movement, if the opponent gets a big chance to increase his win rate, then the AI compares the chosen movement with other movements in order to decide the best movement again. The second method (method-2) is assigning scores on the game board to distinguish good position and bad position. I compared Pure-Monte-Carlo AI(PMCA), method-1 AI, method-2 AI and method-1&2 AI:

|  |  |  |  |
| --- | --- | --- | --- |
| PMCA vs. | method-1 | method-2 | method-1&2 |
| Total Playouts | 20 | 10 | 38 |
| Method Wins | 8 | 6 | 24 |
| Draws | 2 | 1 | 1 |
| PMCA Wins | 10 | 3 | 13 |

Method-1 had about 40% of win rates, and method-2 had 60% of win rates while method-1&2 had about 63% of win rates. From the result, I concluded that method-1 never improves the AI while method-2 has meaningful results because method-2 and method-1&2 have similar win rates. The possible reason that method-1 does not affect the result in the testing is that method-1 and PMCA shared the same Monte-Carlo-Tree database. I guess if they had different databases, then method-1 would work like the min-max algorithm. For method-2, the amount of score was interesting because as changing the score of position, the result was extremely changed.

←For the scores of the positions, I

used this table



The result shows that there is a definitely good position in Othello; therefore the scores need to be carefully decided to actually improve the AI. In conclusion, the second version AI had better performance than the PMCT AI because of method-2 while method-1 does not affect the result; however, actually the performance depends on how the score is assigned at each position because it dramatically changes the result.