TODO: WRITE A NICE TITLE!

Thiago V. de A. Silva
2017719891

December 6, 2017

1 Introduction

Tic-Tac-Toe is a classic 2-player game. Given a 3x3 board, the game takes place in turns, in each turn a player has to choose a blank square in the grid and put his mark on it, whoever gets three marks in a row wins. This game is extensively used as example in artificial intelligence and game theory courses. In this report, we define the player's 1 mark as X and player's 2 as O.

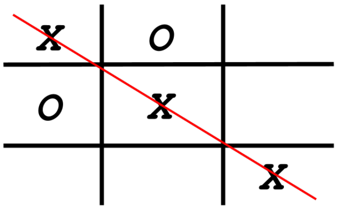

Figure 1: Example of tic-tac-toe game; player 1 won!

There are several variations of the game: 3D tic-tac-toe, Wild tic-tac-toe, Ultimate tic-tac-toe... [1, 3, 2]. On this project, we decided to work with the ultimate version of the game. This variation is interesting because it's simple, there is no clear winning strategies, unlike the classic version, and the number of states is exponentially high.

2 Game Rules

The Ultimate tic-tac-toe is a 2-player game, just like the classic version. Given a 3x3 board, for each square there is a classic tic-tac-toe board, see the figure below for better understanding:

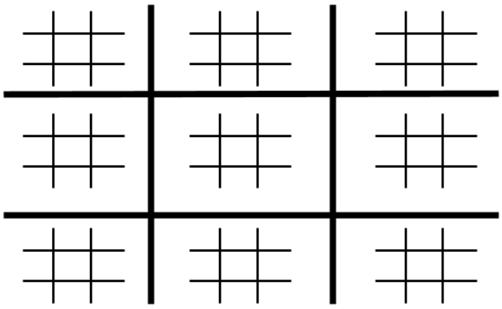

Figure 2: Example of the Ultimate tic-tac-toe game

The game takes place in turns: The first player starts, he chooses any inner board and put his mark in a blank square on position (i, j) of the inner game, then player 2 must play at the (i, j) board. For example: Suppose player 1 chooses the (1, 0) inner game, and marked at position (1, 2); then, player 2 must play the (1, 2) inner game next, in the figure below she played at the position (0, 0) of the (1, 2) inner board, then player 1 must play the (0, 0) inner board, and this game goes on...

The play is defined by four coordinates (x, y, i, j), in which (x, y) indicates the outer position and (i, j) indicates the inner position. Given that the first player played at (x_1, y_1, i_1, j_1), for instance, then player 2 must play at (i_1, j_1, i_2, j_2) next. See the example below:

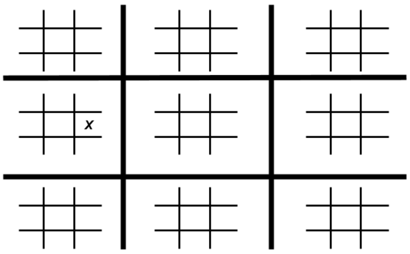

Figure 3: First turn

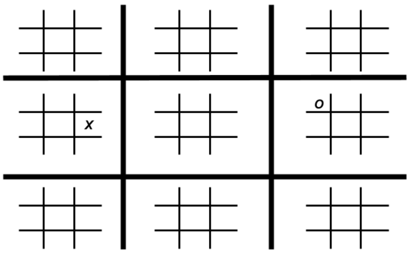

Figure 4: Second turn

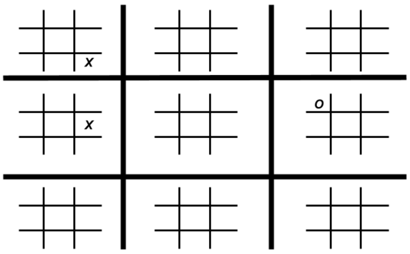

Figure 5: Third turn

3 Related Work

Talk a little bit about the related works.
That paper that the authors showed equivalent classes of the board.
There is also the AI's available online.
And maybe more...

4 Payoff Table

Probably I'm going to define more than one payoff table.
Talk about the tables, why I chose the payoffs described in which one of them.

5 AI

5.1 Random

Talk a little bit about the random strategy.
Show some numbers, like the proportion of player 1 wins, proportion of player 2 wins, proportion of draws.
Average number of moves until the end of the game, median, standard deviation, and everything.

5.2 Alpha-Beta Prunning v1

With the first function for the A^*.

5.3 Alpha-Beta Prunning v2

With the second function I defined for the A^*.

6 Experiments

(1) Show the experiments comparing the results

7 Future Work

(1) For now, there is just the monte carlo tree search.
(2) Maybe create more payoff tables.
(3) Consider using cython, or any optimizer, given that I used python, and used it for simplicity.
(4) Make some pruning using the equivalent classes from that nice paper.

8 Conclusion

References

[1] the free encyclopedia Wikipedia. 3D tic-tac-toe. `https://en.wikipedia.org/wiki/3D_tic-tac-toe`, 2017. [Online; accessed at 06-December-2017].

[2] the free encyclopedia Wikipedia. Ultimate tic-tac-toe. `https://en.wikipedia.org/wiki/Ultimate_tic-tac-toe`, 2017. [Online; accessed at 06-December-2017].

[3] the free encyclopedia Wikipedia. Wild tic-tac-toe. `https://en.wikipedia.org/wiki/Wild_tic-tac-toe`, 2017. [Online; accessed at 06-December-2017].