

1 Introduction

The purpose of this paper is to examine the outcome of a 1000 games played between two versions of the AI. The following are going to be explored:

- Two advanced AI playing against each other
- One advanced AI playing against random AI with the advanced AI starting
- One advanced AI playing against random AI with the random AI starting
- Two random AI playing against each other

2 Methodology

For testing out a 1000 games the following script was used at the end of the file. The only thing that was altered were the lines containing the `computer_move` function.

```
x_victories = 0
o_victories = 0
a = 0
while a != 1000:
    while winner == '':
        computer_move(board, 'X', 'O')
        x_wins = check_cols(board, 'X')[0] + check_rows(board, 'X')[0] + \
            check_diag(board, 'X')[0]
        o_wins = check_cols(board, 'O')[0] + check_rows(board, 'O')[0] + \
            check_diag(board, 'O')[0]
        if x_wins > o_wins:
            x_victories += 1
            a += 1
            break
        if o_wins > x_wins:
            o_victories += 1
            a += 1
            break
        x_wins = o_wins = 0
        computer_move(board, 'O', 'X')
        x_wins = check_cols(board, 'X')[0] + check_rows(board, 'X')[0] + \
            check_diag(board, 'X')[0]
        o_wins = check_cols(board, 'O')[0] + check_rows(board, 'O')[0] + \
            check_diag(board, 'O')[0]
        if x_wins > o_wins:
            x_victories += 1
            a += 1
            break
```

```

if o_wins > x_wins:
    o_victories += 1
    a += 1
    break
x_wins = o_wins = 0
print(a)
board = []
for i in range(rows):
    board.append([])
for j in range(cols):
    board[i].append(' ')
print(x_victories, o_victories)

```

This procedure was performed for size 5×5 , 5×6 , 6×5 , 6×6 , 6×7 , 7×6 and 7×7 boards.

3 Results

X is the checker that always goes first while O goes second.

Advanced AI vs Advanced AI

size	O wins	X wins
5×5	1000	0
6×5	1000	0
5×6	1000	0
6×6	1000	0
6×7	1000	0
7×6	0	1000
7×7	0	1000

Random vs Advanced AI: (random first)

size	O wins	X wins
5×5	36	934
6×5	35	965
5×6	41	959
6×6	29	971
6×7	19	981
7×6	23	977
7×7	16	984

Advanced AI vs Random (AI first)

size	O wins	X wins
5×5	187	813
6×5	143	857
5×6	120	880
6×6	104	896
6×7	78	922
7×6	66	934
7×7	41	951

Random vs Random

size	O wins	X wins
5×5	445	555
6×5	465	535
5×6	435	565
6×6	440	560
6×7	444	556
7×6	473	527
7×7	427	573

4 Discussion

In the first trial with the two advanced AIs would follow the same steps in every trial, so whoever won the first won, would win the following 999. It is interesting to notice that for 5/7 trials the bot that went on the second turn won.

In the random vs AI trials the number of wins for the Advanced AI seemed to increase as the board size increased. This is likely due to a random having to place pieces into more columns, which meant that more of its pieces were separated and the chance of it having a winning combination was decreased. What is interesting to notice here, however, is that when the AI went first, it did way more poorly than when it went second. This combined with the previous observation of 5/7 trials in AI vs AI had the AI that went second win suggests that the AI written is better designed to go second.

Finally, the last table serves to show the advantage of a random bot starting first. As seen in the table, the bot that started first would always have around a 100 more wins than the bot that started second.

Note: I refer to the computer placement function as the AI of the program. That is probably the incorrect vocabulary.