Section 10, Group 1
EG 10112
Prof. Brian Smith

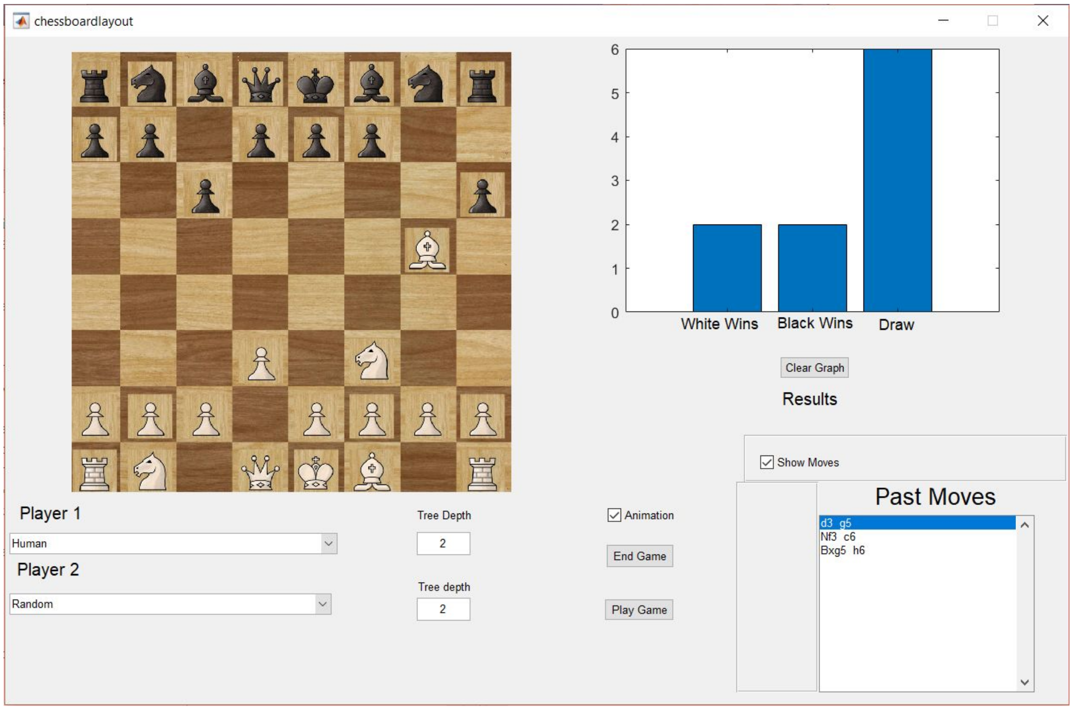
Executive Summary

Background Information & Purpose

The group’s design statement is as follows: Simulate a game of chess, with options for 0, 1, or 2 human players, and compare different strategies for playing the game to determine which is the most effective.

Selected Design

Figure 1 below shows the final working GUI for our project:



Player 1
Human

Tree Depth
2

Player 2
Random

Tree depth
2

☑ Animation

☑ Show Moves

```
d3  g5
Nf3  c6
Bxg5  h6
```

Figure 1. Final Working GUI

As figure 1 shows, the final GUI contains several components. The top left corner has a chess board displaying the current game. It is interactive, and whenever a human is playing, pieces can be dragged to the desired positions. The bottom left has two drop-down menus, one for each player. These menus allow the user to configure who controls white and black, selecting from random, piece addition, piece-square table, hybrid, or human players. There are also text boxes which allow the user to configure how many moves the AI looks ahead. The middle area has some additional controls. The checkboxes allow the user to toggle animations and annotations, and the buttons allow the user to clear the graph and start and end games. The top left corner has a graph of the number of wins and losses, and the bottom left corner shows an annotation of the game.

Results

The group did succeed in fulfilling the design statement on time and to its entirety. We created a fully interactive GUI, which included a display of a chess board, with different options for 0, 1, or 2 human players. When the option for a non-human player was wanted, one could select between different strategies for the computer to use when playing. These strategy options are as follows: Random, Min/Max piece addition, Min/Max piece square, and Min/Max hybrid. In order to find the most successful strategy, we ran 2400 game simulations, testing the different strategies of the computer against each other, and compared the results of each game. The results of our testing can be seen in Figure 2 below:

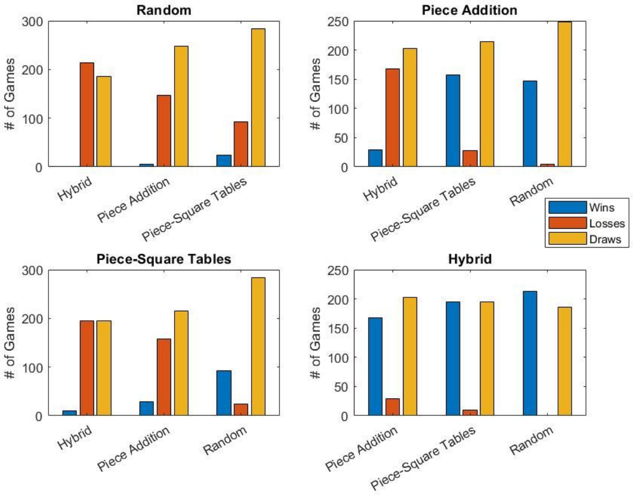


Figure 2. Results of Each Algorithm

Our results showed that overall, Min/Max hybrid was the most successful strategy, followed by Min/Max piece addition, then Min/max piece/square, and finally random was the least successful strategy for the computer to use.

Recommendations

One way the project could be improved optimization of tree depth. Due to how long it takes to run, the tree depth has just been set to a low value. However, processing time can vary wildly depending on the number of next possible moves. If we could somehow optimize the picking of how many moves to look ahead, then the performance of each algorithm would have improved significantly.

Another flaw our project experienced was how it handles the endgame. There are some fairly well-known methods for checkmating depending on what pieces you have, and implementing some of those methods could have been beneficial for our AIs. Games would occur where one AI would get very far ahead of another, but the game would never end because the first AI would never figure out how to checkmate.