PA2 Write-Up

Ryan Rose
A08992549
rrose@ucsd.edu

Clarence Lam
A08678105
c9lam@ucsd.edu

1 Custom Solver

1.1 Name

We named our custom solver AwesomeXpress and our code runs as normal.

1.2 Approach

Our custom solver has the same Minimax and Alpha-Beta pruning algorithms as the Alpha-Beta solver. The difference in our custom solver is that we added a new evaluation function based on the position of the pieces on the board and the number of pieces and kings. Pieces are assigned a weight of 65%, kings are 25%, and being in the center of the board is 10%. The function does the following with these weights: ((.10)(p1Pieces/totalPieces)+(.25)(p1Kings/totalKings) + (.10)(p1Center/totalCenter)) - ((.10)(p2Pieces/totalPieces)+(.25)(p2Kings/totalKings) + (.10)(p2Center/totalCenter)). The algorithm will also pick randomly if there are multiple moves with the same "best" value.

Originally we tried just summing all the pieces on the board times the weight of their position and returning the total value, but this didn't seem to make much difference on play.

1.3 Qualitative Analysis

Overall, all of the algorithms seemed to make some bad decisions when it came to the endgame. The moves are calculated very quickly until a depth limit of 8, but often times the game ends up in a stalemate anyway with the solver moving a king back and forth over the same two spots. To get around this in the custom solver, we added some randomness to how it picks moves with the same "best" value. Other than that weirdness at the end, the solver seemed to work efficiently.

1.4 Custom agent vs Custom Agent

The custom agent was set against itself using several different depth limits and look limits. The depth was set as high 1000 with a look limit of 1000. There was never a problem as long as there was a look limit. With a depth of 10 and no look limit, the agent took 4 to 10 seconds to make a move which is rather slow.

1.5 Custom Agent vs AlphaBeta Agent

The custom agent was set against the AlphaBeta agent with several different depth and look limits. The two agents seemed to switch of winning for the most part and ended up tying for number of wins. They did not tie any matches.

1.6 Custom Agent Average Move Time

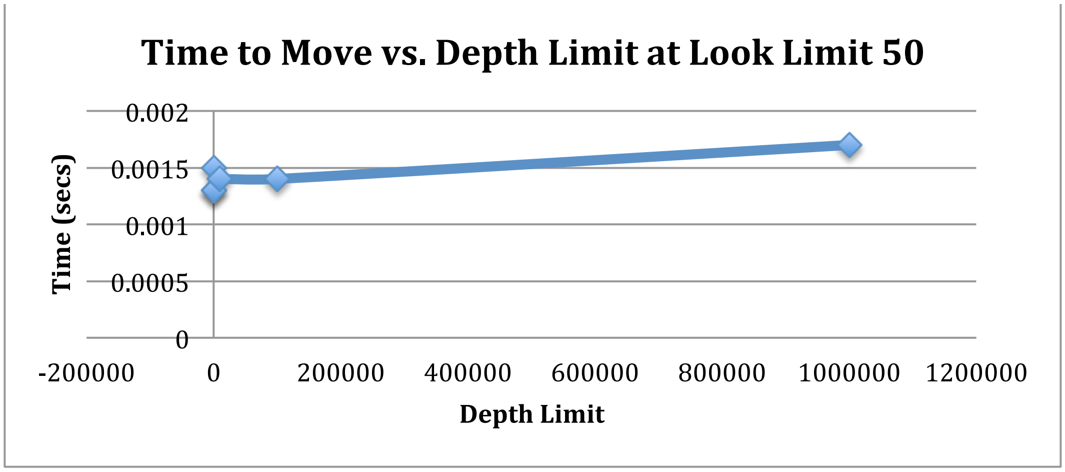

Figure 1: Custom Agent Average Time to Move versus Depth Limit

2 Minimax Agents

2.1 Minimax and AlphaBeta Agents vs Random Agent

Table 1: AlphaBeta agent vs Random agent

Depth, Look Limit	AlphaBeta Wins	Random Wins	Ties
3, 10	80	15	5
4, 10	77	19	4
2, 10	88	22	0
1, 10	75	20	5

Table 2: Minimax agent vs Random agent

Depth, Look Limit	Minimax Wins	Random Wins	Ties
1, 10	63	32	5
2, 10	66	27	7
3, 10	76	20	4
4, 10	72	21	7

2.2 Minimax and AlphaBeta Maximum Depth

The maximum depth is 8 for the Minimax agent to make a move within 1 to 2 seconds.
The maximum depth is 14 for the AlphaBeta agent to make a move within 1 to 2 seconds.
AlphaBeta can search 6 levels deeper than Minimax.

2.2 Minimax Average Time to Move Versus Depth Limit

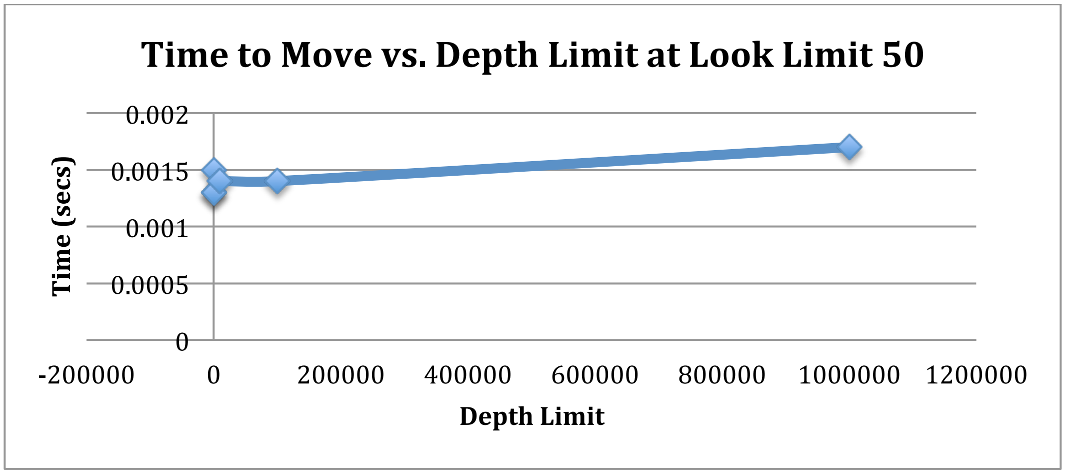

Figure 2: Minimax Agent Average Time to Move versus Depth Limit

2.2 AlphaBeta Average Time to Move Versus Depth Limit

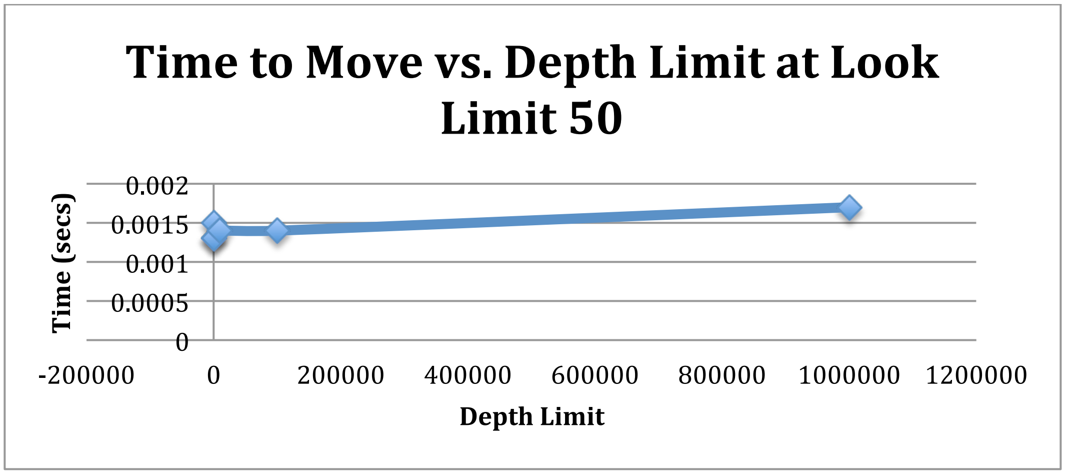

Figure 3: Minimax Agent Average Time to Move versus Depth Limit

3 Author Contributions

We both worked on the project together at the same time. So contributed about equally. We simultaneously ran our finished program on two different computers, with different solvers for p1 and p2 on each computer, in order to quickly collect data. Ryan qualitatively evaluated how our agents played by looking at the code we wrote and by running the agents against one another. Ryan made the graph of average execution time per move as a function of max depth by running each of the three methods with 10, 100, 1,000, 10,000, 100,000, and 1,000,000. Clarence found the maximum depth that a Minimax and Alpha-Beta agent can make a move in 1 or 2 seconds by increasing the depth limit until it the agent took longer than 2 seconds to make a move. Ryan recorded the data for 100 games of Alpha-Beta agent versus Random agent and Clarence recorded the data for 100 games of Custom agent versus Random agent. Clarence recorded the data for Custom agent versus Alpha-beta agent and Ryan recorded the data for Custom agent versus Custom agent.