Relative Importance of Basic Strategy in Othello
Ben Shaman, Agastya Nashier, Josh Lee, Sumant Sharma
Department of Mathematics, Dartmouth College

Abstract
This paper attempts to elucidate the features, and advantages or disadvantages, of a few basic Othello principles. Therefore, we created player classes for an Othello simulation in Python to represent these principles in isolation. We ran simulations of every pairwise combination of these players against each other, playing both black and white, first (n = 10,000 through to the end of the game, and second (n = 320) with Egromatic solver implemented starting at move 45. This data allowed us to reconstruct payoff matrices for the strategies played against each other and create dynamic visualizations. Further analysis provides explanations for certain elements of conventional Othello wisdom, including the topics of corners and parity. Our work demonstrates and analyzes the complex dynamics of a relatively simple game, from which new players can learn a lot.

Methods

Player Characteristics
To test various basic strategy principles, we created five different player classes, each intended to isolate an element of Othello's strategy.

1. Random Player
   - Plays a random legal move on the board.
   - Random player acts as our control class - a control to compare other strategies to.

2. Greedy Player
   - Plays the move that gains the most discs on every turn.
   - This strategy is the intuitive choice for many beginning players; however, it only leads to short-term gains that actually weaken the player's position. The short-term gains of this strategy are easily counteracted in the long run.

3. Directional Player
   - Plays the move that flips the fewest number of possible directions.
   - A pitfall that some players fall into is making moves that are too "loud". Moves that flip in many directions have far more hard-to-anticipate rippling consequences, so minimizing the number of directions can be beneficial in many situations, especially in the early game.

4. Defensive Player
   - Plays the move that maximizes the difference between players' and opponents' number of legal moves.
   - One of the most effective ways of controlling the outcome of Othello is to give your opponent a very limited number of legal possible moves. From there, you can force them to make moves that benefit your position.

5. Static Square Preference Player
   - Plays based on a pre-set grid of preferences for certain squares. In our model, each square is assigned a value, and the move's score is calculated by $2x^2$ the square being placed on plus the values of squares being flipped.
   - Most squares have negative values to avoid the greedy player's mistakes. The rationale is that some squares, like corners, are extremely advantageous to possess, whereas others could benefit the opponent, so moves should be rewarded and penalized accordingly.

Simulation
To simulate these strategies and reconstruct payoff matrices, we:
- Created an Othello program in Python.
- Played every pairwise combination of the five players against each other through to the end of the game, 10,000 times for each color.
- Since endgame strategy differs significantly from the rest of the game, we performed a second simulation using Egromatic, an Othello solver, to play perfectly from move 45 to the end of the game. We were able to run this simulation 320 times for every combination.
- From these simulations, we extracted and analyzed the win-lose percentage of each matchup as well as the margin of victory by percentage of discs held.

Simulation Results

<table>
<thead>
<tr>
<th>Player/Player</th>
<th>Random</th>
<th>Greedy</th>
<th>Direction</th>
<th>Defensive</th>
<th>Preference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0.5</td>
<td>0.3999</td>
<td>0.3474</td>
<td>0.2532</td>
<td>0.0347</td>
</tr>
<tr>
<td>Greedy</td>
<td>0.6151</td>
<td>0.4</td>
<td>0.55925</td>
<td>0.3545</td>
<td>0.64475</td>
</tr>
<tr>
<td>Direction</td>
<td>0.63195</td>
<td>0.44037</td>
<td>0.3</td>
<td>0.37137</td>
<td>0.33875</td>
</tr>
<tr>
<td>Defensive</td>
<td>0.7462</td>
<td>0.45982</td>
<td>0.62843</td>
<td>0.3</td>
<td>0.20485</td>
</tr>
<tr>
<td>Preference</td>
<td>0.86225</td>
<td>0.83582</td>
<td>0.84612</td>
<td>0.79195</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Further Analysis
Looking beyond these player strategies, there is the game theoretical question of whether it is better to play black or white. Figure 2 shows an example of a typical early endgame position in Othello. A common pattern is the occurrence of sectioned off regions of open squares. This leads to a parity advantage for white, who goes second. In other words, if they play intelligently, they will play the last move in each region, gaining the maximum possible number of discs. When played through to the end, the greedy strategy, by nature of constantly trying to maximize piece count in the short term, gives away this parity advantage. That is why in Table 3a, greedy plays better against itself when playing black. With the endgame solver, all players have an advantage playing white.

Figure 1: Hypothetical Othello position

Figure 2: Dynamo 35 simulation without endgame solver (n=10000, left) and with endgame solver (n=320, right)

Table 2: Payoff matrix for 5 strategies with win percentage, played through move 45, then Egromatic for the endgame.

<table>
<thead>
<tr>
<th>Player/Player</th>
<th>Random</th>
<th>Greedy</th>
<th>Direction</th>
<th>Defensive</th>
<th>Preference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0.5</td>
<td>0.70435</td>
<td>0.5664</td>
<td>0.17417</td>
<td>0.0379</td>
</tr>
<tr>
<td>Greedy</td>
<td>0.29354</td>
<td>0.5</td>
<td>0.3232</td>
<td>0.2185</td>
<td>0.0215</td>
</tr>
<tr>
<td>Direction</td>
<td>0.6334</td>
<td>0.6677</td>
<td>0.3</td>
<td>0.0485</td>
<td>0.0345</td>
</tr>
<tr>
<td>Defensive</td>
<td>0.82928</td>
<td>0.87285</td>
<td>0.6955</td>
<td>0.5725</td>
<td>0.2325</td>
</tr>
<tr>
<td>Preference</td>
<td>0.9217</td>
<td>0.97885</td>
<td>0.9655</td>
<td>0.7625</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Table 3: (a) White players' advantage without endgame solver, (b) with endgame solver

Limitations and Future Research
One limitation we face is that our analysis only features $n = 320$ iterations for our endgame solver, so the results are relevant, but may not quite as robust as we want. Another is that we have not yet implemented any look-ahead for our players. The static square preference player dominates in this environment because its preloaded information contains far more game knowledge than any of the others could possibly access. It is preprogrammed to avoid some of the most egregious blunders, whereas the others are not. We hypothesize that implementing even basic 3-ply look-ahead could reduce this advantage, swinging in favor of the defensive player. Alternatively, we could add a level of "dist-proofing" to the other strategies. Future research could include determining which strategies benefit most from look-ahead as well as combining elements of multiple strategies. We could also tweak the criteria by which players such as defensive and direction minimizing calculate their optimal moves.

Conclusion
The main implication of our findings is that this type of analysis can be very helpful for the understanding of beginning players. Despite Takaizawa's claim to have weakly solved Othello (2024), this analysis shows that there is still much we can learn and confirm from a more rudimentary analysis of the game's strategies.

References
[1] Takaiizawa for Console (endgame solver) by Takuto Yamada (updated 2024)

Thank you to Jared Heath '25 for his help implementing and debugging Egromatic.