B552 Project Sketch
March 16, 2021

1. A more Human Chess Algorithm: Using Case-Based Reasoning for Holistic Search Space Pruning in Chess

2. Zachary Wilkerson, Rowan Lavelle, Josep Han

3. Most simple chess algorithms (i.e., minimax-based ones) have to navigate a significant search space, which often makes it computationally infeasible to consider positions more than a few moves in advance; by contrast, humans often "rule out" substantial majorities of this search space as unreasonable/pointless moves. We propose using case-based reasoning based on "positional remindings" (probably combined with some rule-based approaches) to prune the search space, which should significantly augment a minimax-based approach.

4. This project can act as proxy for using case-based reasoning to help address problems with large branching factors in such a way that search may still be considered as a feasible solution. Indirectly, this will also be interesting in the context of a hybrid system applied to games.

5. We envision the UI as an ASCII-based interface that will be able to accommodate algebraic notation to define moves (e.g., Ne4, Kf2, etc.). This system will be modified slightly from existing code to allow for castling, check, checkmate, en passant, and under-promotion. The player (or players—we will likely consider evaluation against other algorithms, see below) will be defined in terms of object-oriented design. We will all three work on the implementation, scaffolding, and optimization for these items, building off of our B551 code.

   We envision that the reasoner's cases will be based significantly on rules of thumb (e.g., when the king has castled, it is unwise to move any of the three pawns in front of it), with the ability to apply adaptation to generate exceptions to these rules (e.g., one can take an opponent's queen with one of those three pawns in this specific position, giving me a huge material advantage).

6. This was somewhat described above, but we intend to apply the following. Since these areas are interrelated, we will be helping each other out on all aspects of the project, but Zach will generally focus on 1,2; Rowan will generally focus on 3,4; Josep will generally focus on 3,5 (these may update in the future; as time permits, we will work toward all of these):
   1. case-based reasoning (and likely rule-based reasoning as well for more absolute concepts like check and checkmate) to emulate holistic rules of thumb for matching positions during play
   2. adaptation rules to apply to these rules/cases based on the exact board state (i.e., when exceptions to those rules are required).

3. a detailed rules-based/case-based system for openings in particular (the early moves in chess are well-studied and specific patterns exist in high levels of play)
4. potential exploration of dynamic reasoning (i.e., updating heuristics/weights/ similarity metrics) based on playing experience.
5. evaluation of board states using positional information
    1. Advantages over the course of the game (post-game evaluation)
    2. Consequences of a move (post-move evaluation)

7. We will likely use a two-pronged evaluation technique. First and foremost, this program needs to be evaluated against other players; we envision at least 3 such players: one that makes random moves, one that uses minimax alone (and due to time constraints may only be allowed to look a couple moves in advance), and us as human players (we should be able to beat a limited minimax player most of the time). Along with this, it will also be interesting to explore how much of a pruning improvement that applying case-based reasoning allows (e.g., can it double the number of moves in advance that a minimax algorithm can look at in the same amount of time?). That is, we feel like the metric of success will be a combination of how effectively the search space is pruned (which we can evaluate directly) intersected with its overall playing performance (which will be measured more holistically). If this metric is unsuitable for self-evaluation (see 6.5), we may also consider success relative to chess puzzles or similar techniques.

8. We expect that adaptation/dynamics will be very challenging. In chess, there are countless “tricks of the trade” that players use to recognize patterns in positions and anticipate the most likely outcome for many moves in advance; however very few of these ideas are without exception, so we think adaptation and general algorithm dynamics will be simultaneously very important and very difficult as a means of preventing the algorithm from making any rules of thumb absolute while still allowing pruning to actually happen. Furthermore, it will also be challenging to develop an efficient indexing system that can quickly analyze the board and identify cases that apply in the given position (e.g., without having to analyze the entire board repeatedly for each potential case).

9. We hope to demonstrate a strong grasp both of case-based reasoning as a concept as well as our ability to generate and apply rule-based systems to a concrete problem. We also plan to apply strong team-based programming skills to navigate a project with three group members. We hope to learn more about adaptation in case-based reasoning generally (Zach’s current work to date has focused heavily on retrieval), and we look forward to exploring possible interplay between case-based reasoning and search algorithms.