2. Background and Research

# 1. Chess as a game

## a. Rules and difficulties

Chess as a game has been around for almost two centuries[[1]](#footnote-1) in different versions. However, the most standard version nowadays, dates back to the final decades of the 15th century[[2]](#footnote-2). In the early years of the game, special moves such as castling or pawn opening double moves did not yet exist, but were later introduced into the game, which saw different rules (such as letting either white or black move first) but preserved the essence until the 19th century, when all of the current rules were finally fixated.

Some of these special rules increase the game complexity. Their main purpose is avoiding certain loopholes, such as the possibility of matches that would last for an infinite number of moves. While being greatly useful to enhance the possibilities and strategies in human vs human matches, they do make it a lot harder to develop artificial intelligence engines suitable for this game, as well as making it more complex to evaluate instances of the board, if all special rules are taken into account.

The main rules to take into account are Castling, En-Passant, Pawn Promotion, draw by Stalemate (formerly considered to be a win condition and still argued by some[[3]](#footnote-3)), Fifty-Move rule and Threefold repetition of position.

## b. Past attempts to make a chess AI

Chess is notorious for being a very computationally intensive game, having exponential complexity in n (on an board) to find an optimal strategy[[4]](#footnote-4). This means that for a given starting board, there is an estimate of possible different games of chess that could be legally played[[5]](#footnote-5). This number is known as *Shannon Number* and it was an attempt made by *Claude Shannon* to demonstrate that if someone were to create a computer that played all possible chess games until an end in order to make a decision on the move to make next, this computer would never be able to make a move, therefore suggesting that brute-force approaches to creating a chess engine should be abandoned.

# 2. Technical knowledge

## a. Algorithms

### i. MinMax

### ii. NegMax (Not used in the project)

### iii. Alpha Beta Pruning

### iv. Monte Carlo Tree Search

### v. Neural Networks (Not used in the project)

## b. Design Patterns

## c. Software Engineering Practices

### i. Documentation

### ii. Coding

### iii. Debugging

### iv. Testing

#### 1. Unit Tests

#### 2. Alpha Releases and Testers

#### 3. Nim (Only objective research on this game)

#### 4. Chess End Games

## d. Game API

### i. Description

### ii. Original API

### iii. Features

### iv. Changes

1. Niklesh Kumar Jain (2014), “Hindi and the Origins of Chess”, Chessbase.com. [↑](#footnote-ref-1)
2. Hooper & Whyld (1992), “The Oxford Companion to Chess”, pp. 173-175. [↑](#footnote-ref-2)
3. Larry Kaufman (Sept 2009): “Calling stalemate a draw is completely illogical, since it represents the ultimate zugzwang, where any move would get your king taken.” [↑](#footnote-ref-3)
4. Fraenkel & Lichtenstein (1981), “Computing a perfect strategy for nxn chess requires time exponential in n”, Journal of Combinatorial Theory, Series A (31): pp. 199-214. [↑](#footnote-ref-4)
5. Claude Shannon (1950), “Programming a Computer for Playing Chess”, Philosophical Magazine. 41 (314) [↑](#footnote-ref-5)