**GAME OF CHESS – SABER**

**GROUP-12**

**Graphical user interface, application, Teams

Description automatically generated**

**INTRODUCTION:**

Artificial Intelligence is vast in current time. There were many ideas of projects to choose from like facial recognition, chat bot projects but our group decided to go for the GAME OF CHESS as it is most common game played throughout. Most people playing chess has issue like they are single player, but chess is a multiplayer game, so we decided to take help AI to solve this issue. The AI designed provides a good competition to the player as the machine plays such an immense moves within a short period of time. Our chess game agent-based model is based on Omniscient agent. We worked on rational agent as it will work according to the precepts and knowledge and it tries to maximize its performance and plays with an immense moves which is hard for a human player to implement normally within the short period of time.

**BACKGROUND AND RELATED WORK:**

There was a latest research paper published by google regarding their latest AI evolution, which is AlphaZero, that is our main motivation. It has developed superhuman performance in chess and had mastered its rules in just four hours. In an actual competition with world chess champion Stockfish, AlphaZero won 25 games out of 100 and 75 games were draw therefore no wins for Stockfish and no defeats for AlphaZero.

**Reference:**

SeventhQueen. (2018, January 29). How Google's AI mastered all chess knowledge in just 4 hours. Retrieved February 13, 2021, from

<https://www.asianentrepreneur.org/googles-ai-mastered-chess-knowledge-just-4-hours/>.

In an essay for The New York Times, famous mathematician Steven Strogatz praised the recently published performance results of AlphaZero, the board game–playing AI developed by DeepMind, a British AI company acquired by Google in 2014. While his examination of AlphaZero’s findings is an interesting read, some of the conclusions Strogatz draws about the general advances in AI are problematic. [AlphaZero] clearly displays a breed of intellect that humans have not seen before, and that we will be mulling over for a long time to come says Strogatz. Also, by playing against itself and updating its neural network as it learned from experience, AlphaZero discovered the principles of chess on its own and quickly became the best player ever.

**Reference:**

B., By, Dickson, B., -, Ben DicksonBen is a software engineer and the founder of TechTalks. He writes about technology; Ben is a software engineer and the founder of TechTalks. He writes about technology, & Var block\_td\_uid\_6\_60283be41a07b = newtdBlock();block\_td\_uid\_6\_60283be41a07b.id ="td\_uid\_6\_60283be41a07b";block\_td\_uid\_6\_60283be41a07b.atts = '{"limit":6. (2019,September 25). The problem with anthropomorphizing artificial intelligence. Retrieved February 13, 2021, from link:

<https://bdtechtalks.com/2019/01/02/humanizing-ai-deep-learning-alphazero/>.

**ILLUSTRATION / FIGURE:**

The name of our AI is Saber as Saber was the king of knights and chess is basically the game of King. Our agent-based model of chess board is as:

A picture containing diagram

Description automatically generated

In the search depth drop down the player can have options from 1-5 which is the depth of the tree. The position evaluated shows the position evaluation of the AI agent and time is time taken by the AI to perform a particular move. This time per move changes according to the depth of the tree as in if the depth of the tree is more it will take more time to search and therefore time taken to perform a move by AI will increase. The last position shows the two positions where first is the position by AI and the second is the position performed by the human player.

Text

Description automatically generated

**SIMULATION DETAILS:**

In this game, every agent has their unique steps. The main concept is to implement pieces of chess and the ways it can make moves. We have divided the project in four parts where the first two were to make board, pieces, generate moves and board and position evaluation. The next step was to implement Minmax Algorithm and then Alpha-Beta pruning. We have developed the project using java scripting, jQuery, and HTML-CSS for creating the board and generate moves. Here, we used chess.js for generating moves of all the pieces and chessboard.js for the board. The chess move generation library defines all the rules in the game. The task was to compute all legal movements for the given board state. The easiest method for position evaluation is to use table to count the relative power of pieces on the board. Here, we used the evaluation function which constructs an algorithm that selects the step with the best evaluation. Then we used minmax algorithm to construct a search tree from which the algorithm could select the best course of action. The recursive tree of all the possible moves is explored to given depth entered by the user and the location is evaluated and after that based on whether it’s a white or black to pass, the child’s smallest or largest value is returned to the parent node. Then we used the alpha-beta pruning to make the move implemented by the AI much smoother and more effective.

**DISCUSSION AND OBSERVATION:**

In our model we did the bug fixing and the evaluation as we count the content that is found on the board which makes our initial measurement function very naïve. Our model shows all the possible moves a piece can have when the player puts his cursor on the piece and if the player plays an invalid move, then the piece returns to its original position which says that our algorithm is more efficient and precise. Our model works precisely but there are much more improvements like move ordering, much faster generation of moves, implement a timer for particular moves, and if possible, end game specification we are willing to make in the future.

**Link To our Project :**

<http://chovatik.myweb.cs.uwindsor.ca/Saber-game%20of%20chess/index.html>