

CS 404 ASSIGNMENT3

Ali Koray Cankı 26871

21.05.2023

1 Loopless Slant Game Definition

Loopless Slant Game is a Human versus AI turn-based game where played on a $m \times m$ matrix. The aim of the game is to mark a cell with diagonal lines such that the sum of the neighboring cells equals the number in its corresponding circled value. If it can be done successfully increment the player's score and decrease the opponent's score corresponding to the circled value. "/" diagonal line is for Humans and "\" is for AI.

2 Game Tree Search Model

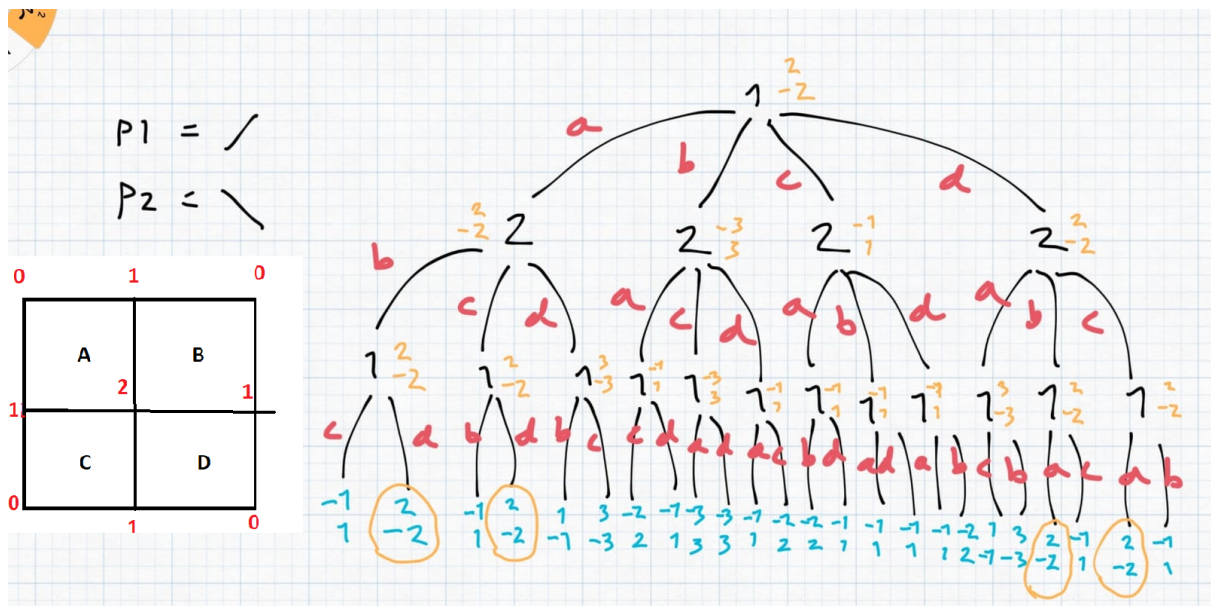


Figure 1: Game Tree Search Model Representation of Corresponding Grid

2.1 Players

There are two players in this game. Player 1 is a human player and Player 2 is an AI player.

2.2 The States

A state in this game is composed of three main components. The first component is a two-dimensional array representing the current configuration of the game grid, with each cell either empty or marked by one player. And the grid shows the position of each player's move. Secondly, there are the scores for both players which take the form as integers reflecting their points total thus far in the play. Lastly, there is an indication for whose turn it currently is; taking on the binary form (0 or 1) to specify if it's Player 0's turn or Player 1's. As it can be seen from the image the nodes of the tree represent the states.

2.3 The Initial State

The initial state of the game is a game grid where all of the cells of the grid are empty, both players' scores are zero, and it's Player 1's turn.

2.4 The Terminal State

When all cells in the game are marked and there are no valid moves left, The game has reached the terminal state, it ends and shows the final scores of each player.

2.5 State Transition Function

The state transition function takes the current state and move to a new position (a pair of x,y coordinates indicating where to mark on the grid if the position of the x,y coordinates are valid), and it returns a new state. The new state is reached when the corresponding (x,y) coordinates of the grid are marked and it updates the scores based on the evaluation of the move and it switches to turn the other player. In my code, the `make_move` function changes the state, if the move is valid it marks the cell and by calling the `update_score` function it updates the new score and finally changes the user's turn.

2.6 Payoff Function

Payoff Function: The evaluate method of the game is used to work out which state of play is best for each player. For Player 1 (the human), a high score difference would be good, while for Player 2 (the AI) it would be better if the scores were closer together. The evaluate method looks at these differences and decides how favourable that particular outcome is to both players.

2.7 Alpha Beta Pruning

Alpha-beta pruning is an optimization method used for searching game trees. It involves avoiding paths that won't result in a favorable outcome, giving the search algorithm more time to evaluate only essential parts of the tree. By doing so, it helps speed up searches and makes them more efficient as irrelevant branches are skipped over.

2.8 Depth Search

The depth of the tree is limited to 3 because to control the difficulty of computations. For examples it takes too much time to compute the whole depth of 4x4 grid so we limit the searching depth.

3 The Conclusion

This game utilizes artificial intelligence to create an interesting and exciting gaming experience. The Alpha-Beta pruning algorithm is used by the AI player which allows it to consider all possibilities according to the current state of the game before making a move. This results in a dynamic and adaptive AI that can analyze moves efficiently, outmatching players with its strategic ability. Through this example, we see how AI can be applied in gaming development; providing gamers with competitive opponents while also adding depth and complexity into their gameplay experiences.