Project Outline

Overview

Philosophers' Football (or Phutball), named after the famous Monty Python sketch, is a board game invented by the brilliant John Conway, who sadly passed in the pandemic earlier this year.

In this project, we will build software for playing this game. There will be three versions:

- a version where a non-human player chooses its moves at random;
- a version where a non-human player chooses its moves using the minimax algorithm; and
- a version where a non-human player chooses its moves using reinforcement learning.

Random

Each possible new state of the board will be generated and one will be picked at random. This will involve going through the board and seeing if it is possible to place a new player down or kick the ball using players along a line. One of these new board states are chosen at random, making this new state the main board's state.

Minimax

The minimax algorithm will use the generated new board states to build trees of subsequent moves. Like the random version, it will need all possible moves from the current position, but it will then get all the possible moves of those positions and so on until we get to the end (all paths explored) or we stop at a certain depth.

Interesting sized boards (ones which require more than 2 moves to win) would prove too large to search to the bottom of the tree, as the trees would grow exponentially. So we will use a heuristic on the potential board after a certain depth to determine if a player should take that path or not. The heuristic will look at that board's state and assign it a value depending on certain factors, such as the ball position or number of players on a side of the board.

To further cut down on execution time, we will use Alpha–beta pruning to skip over trees which would not harbour better results for either the minimiser/maximiser. As the maximiser tries to maximise their score, they would not pick an option which would be better for the minimiser, so there's no point exploring that tree as it would never choose it, and vice versa.

Reinforcement Learning

Through the use of reinforcement learning, we will craft an AI which knows what moves are favourable and ones which are not. This is done by rewarding the AI when it wins a game and penalising it when it loses a game.

The idea is to train the AI to come up with it's own strategies instead of programming what it should do in a situation. We give it the capacity to recall situations which are similar to the one it currently faces and if it was rewarded or penalised on what actions it took before. Through trial-and-error, the AI will get a feel of what moves are good to play and which one should be avoided.