Monopoly Playing Program

1 Problem Definition

This project aims to build an agent which plays the classic board game monopoly with one or more players. An adjudicator/simulator program will allow us humans to view the stages of the game play and serve as a platform for the agent to communicate with the game.

2 Literature Review

We studied existing research done on game theory, reinforcement learning and building an agent for monopoly and similar games.

2.1 Generating interesting Monopoly boards from open data, Marie Gustafsson Friberger et al., 2012 IEEE Conference on Computational Intelligence and Games [1]

This paper explores the possibility of working with data outside a monopoly board game and transforming it into a board game. The authors built a Java based tool called Open Data Monopoly which uses real public data from the UK Government's website and converts it into a board game just like monopoly. The program extracts indicator data for geographical locations and take indicators and their weights from the user. It calculates prosperity for each location between -1 and 1. In order to map the streets on the game, it asks the user to weigh prosperity distributions, geographical distribution and notability (based on Wikipedia article length). Based on certain formulae, street selection is done and the community/chest cards are created.

We would like to reverse engineer the formulae used in this program to compare with our standard game and make comments about the meta-characteristics of the properties.

2.2 Luck, Logic and White Lies, Jorg Bewersdorff [3]

In his book *Luck, Logic and White Lies* Jörg Bewersdorff explores the game of monopoly using Markov Chains defining each square as a state. The key idea was to prove the chances of landing on a square on a monopoly board can be expressed as a regular Markov Chain and in the end has a fixed probability that would depend on the last one, two or three events in the game. The varying event dependency is introduced because of the rule of double rolls that is specific to monopoly and the introduction of chance and community chest cards.

2.3 Breakdown of Monopoly - The Game [4]

In her book, "The Indisputable Existence of Santa Claus", Dr Hannah Fry builds on the idea of expressing monopoly sites as Markov Chains. While the probability of reaching each site is given importance, she's further explored how things change once money is brought into the picture and gives a basic idea of how the decisions would change once all the rules of the game are introduced. To create a new ranking of different sites, she uses ROI as a metric and calculates how many opponent turns would each site take to return the full investment and start making profit. Refer Fig. 1.1.

A further build up on this metric is by including a new rule about colour sets into the calculation and re-evaluating the ranking of sites based on the colour. Refer Fig. 1.2

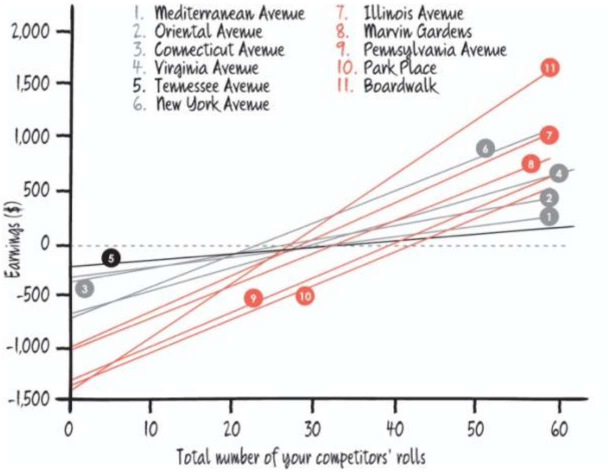

Fig 1.1: Total number of competitor's rolls versus Earnings

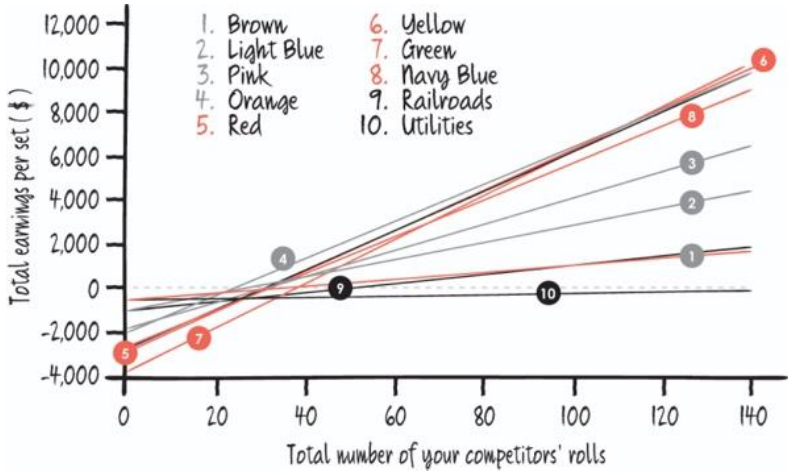

Fig 1.2: Total number of competitor's rolls versus Total Earnings per Set

While this is good, winning probability is also affected by the houses and hotels that can be built on the site and calculating the ROI on those is equally important. The fastest return on investment according to the calculations is to get 3 houses per set. Refer Fig. 1.3.

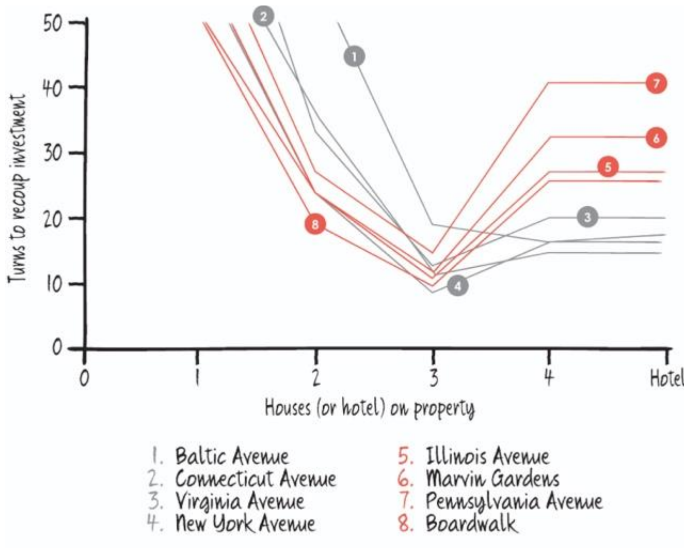

Fig 1.3: Houses on Property versus Turns to recoup the Investment

2.4 Multi-Agent Reinforcement Learning: A Survey, Lucian Busoniu et al., 2006 9th International Conference on Control, Automation, Robotics and Vision [2]

The authors have conducted a survey on different Multi-Agent Reinforcement Learning models in this publication. They have reviewed the challenges of multi-agent reinforcement learning [5], the methods to address them, and we have provided specific conclusions and open issues for each class of methods.

We intend to look at this paper closely and choose a learning method which best will suit our optimization function to build the agent.

3 Data Collection

We were able to discover a few datasets which were generated to study the rudimentary statistics of the game. We plan to create a program which will simulate die rolls and thus allow us to dig deeper into the probability functions of landing on certain squares or the return on investment for each property. [9], [10] are some sample datasets we will be using to start an analysis of these statistics.

For the most part, we will be generating a lot of the data required using the simulator program to analyse the game play. A non-exhaustive list of features we will try to look at for each run are:

1. Turn Number, Player Number
2. Card Drawn (if any)
3. Die rolls and their sum
4. Player Position on the Board
5. Site Information
6. Rent Owed and to whom
7. Rent Received and by whom

We also intend to study games won by champions in detail, try to map a strategy they follow as an algorithm and see if we can find any patterns in them. Some of these game plays and player interviews are publicly available on fandom websites and YouTube.

4 Preliminary Work

Building on the idea from Section 2.3, we found a simulator [11] which we used to generate the probabilities of landing on each site using a million simulations of a game of 100 rounds each. We ran this simulator and mapped the probability of landing on each square. These probabilities are represented in the figure below.

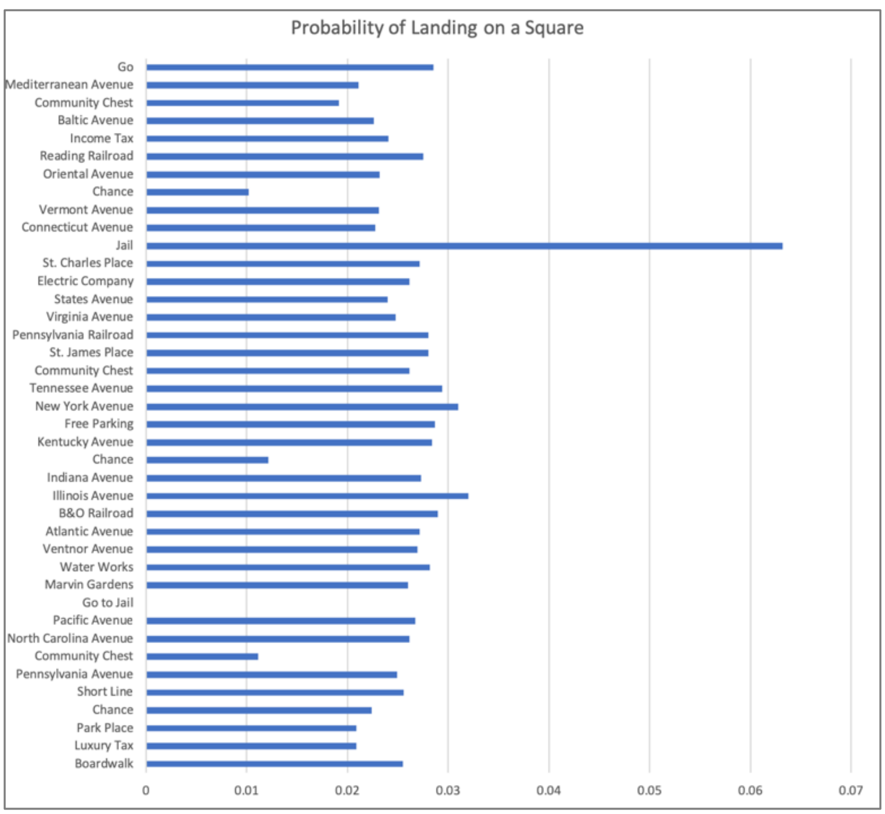

Fig 2: Probability of landing on a Monopoly Board Square

Ranking squares based on the probability of players landing on each of them in a particular turn is a good metric for considering importance of any property on the board but is quite limited as it does not take into account a lot of other factors of the game itself like, how these rankings would be affected when money is introduced in the game. Also, how to use these rankings where there are multiple players playing the game.

We realized that there are certain aspects of the game which we will pay closer attention to in order to make the decisions on each stage. Some of these aspects are:

1. Distribution of money with time for the player who bankrupts first, an average player and the winner to predict vulnerable players.
2. Probability of landing on each square to help decide if the agent should buy that property or not.
3. Return On Investment for each property to help decide if the agent should buy that property or not.
4. Probability that a Chance Card and/or Community Card will be drawn.
5. Where do Chance and Community Card tend to place you on the board.
6. Probability of landing in jail and when is it beneficial during the game.
7. Given x amount of money, should the agent build houses & hotels or buy more property.
8. A shrewd idea humans implement: should you stop at building 3 houses and restrict the supply of houses once you are in the lead of the game. Is this a good strategy or can it backfire at the agent?
9. To be able to build houses/hotels, one needs to hold properties of the same colour. We want to evaluate the probability this can be achieved and a function to maximize the chances of this.
10. Trading and how it affects the game play.
11. Probability of your die score (directly affects your position)

We have started gathering statistics on these above points and we will continue to do so till we find some reasonable strategies to play the game.

5 Challenges

The key challenges in this project are:

1. Building a common simulator and adjudicator with other teams to judge the performance of the agent and play agents off each other.
2. Creating a large enough dataset to train the agent and improve its game play.
3. Figuring out the right attributes for the dataset and the appropriate modelling.
4. Picking the apt trade-offs for the agent to gain monopoly in the game.
    a. Strike the right balance between chance and strategy of the game.
5. Gain a sense of intuition for the robot (like a human) to make the right decision.

6 Project Plan

We will take this project up in the following stages,

1. Starting with a baseline model which will be using a greedy approach to buying properties i.e. it will buy everything that it lands on and can afford.
2. Defining a scoring function over the sites which will improve the decision making ability of our agent regarding purchasing of various sites across the board.
3. Including various attributes like increased ROI from color sets, houses/hotels as compared to individual sites.
4. Further improving decisions based on game state and decisions made by other players using reinforced learning [5].

To track the progress report and the final report about this project, here is a list of tasks we will try to achieve by the end of these milestones:

Mid-Project Progress Report:

1. Build a adjudicator/simulator collaborating with other teams.
2. Have a baseline agent ready with some improvements over it.

Final Project Report:

1. A ready to play agent with bots of other teams fully trained on the built dataset.

7 References

1. Generating interesting Monopoly boards from open data, Marie Gustafsson Friberger et al., 2012 IEEE Conference on Computational Intelligence and Games [1]
2. Bewersdoff, Jorg, *Luck, Logic and white lies,* A K Peters, Wesley Massachusetts [2]
3. Monte-Carlo Tree Search: A New Framework for Game AI, Guillaume Chaslot et al., Proceedings of the Fourth Artificial Intelligence and Interactive Digital Entertainment Conference [3]
4. Evolutionary game theory and multi-agent reinforcement learning, Karl Tuyls and Ann Nowe, The Knowledge Engineering Review [4]
5. Multi-Agent Reinforcement Learning: A Survey, Lucian Busoniu et al., 2006 9th International Conference on Control, Automation, Robotics and Vision [5]
6. Here’s how to win at Monopoly, according to math experts, Hannah Fry and Thomas Oléron [6]
7. Hacking Monopoly! [7]
8. Computer Game Programming II: AI, University of Southern Denmark [8]
9. Monopoly Board Frequencies and Economics, data set [9]
10. Monopoly Simulations, koaning.io [10]
11. Reference code for Monopoly frequencies [11]