AI for Gaia Project

Team Gaia Project: Yifan Zhou, Zheng Li
Project Sponsor: Jon Rahoi

UNIVERSITY OF SAN FRANCISCO

Gaia Project Board Game

In Gaia Project, each player controls one of 14 factions striving to peacefully colonize the Terra Mystica galaxy. Each faction has different environmental needs for surviving on planets. Therefore, the factions have mastered *terraforming*, enabling them to make different planet types habitable for themselves.

During the game, the player will take actions to get as many victory points as possible, such as building mines, upgrading structures, doing research and forming federations, among others.

In the end, after six rounds of play, the faction with the most victory points wins.

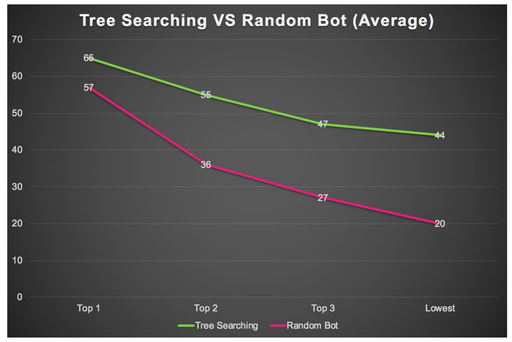
Approaches

1. Random Bot

This AI makes all decisions at random. This means that we can not ensure it can win a game, just finish. The algorithm has poor performance; however, building a Random Bot made us more familiar with the game logic and the new language (Typescript), and gave us a baseline for evaluating the performance of other algorithms, so it was still useful to implement it.

2. Tree Searching

To reach a reasonable compromise between running time and AI performance, we built a one-ply search tree. This AI will not choose randomly. It may not make the best choice in all actions, but it goes through all possible choices and tries to maximize victory points by exploring a search tree one level deep. In other words, it's a *shallow, greedy* search.



3. Neural Networks

In order to improve the performance of our AI, we tried to build a neural network and train it by using the data from tree searching. We save all game states and extract all data, and then feed them as values to the input layer of the neural network. After that, by using **backpropagation** (a method used to calculate a gradient in the calculation of weights), **feed forward** through the network to generate the output values, and calculate the **loss/cost** function (error term), then get the deltas of all output and hidden neurons. Later, the network can update the weights by multiplying the weight's output delta and input activation and giving a ratio of the gradient of weight. Finally, the neural network can "learn" by repeating the above processes.

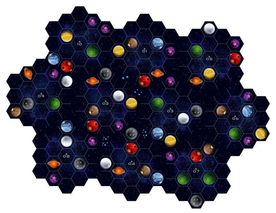
Tools

- **React:** A frontend Javascript framework that we used for testing the basic frontend of the project.
- **Npm Libraries**: A package manager for the JavaScript programming language.
- **TypeScript**: A superset of JavaScript; the main language we used in this project.
- **Node.js**: A JavaScript run-time environment that can execute JavaScript code outside of a browser.
- **TensorFlow**: An open-source machine learning library that provides a Python API.

Game Rules

Races and Planets Type

1. Each player chooses a faction (race) at the beginning of the game, and they will get a matching faction board which has the following initial resources: 3 knowledge, 4 ore, and 15 credits. Then, each player can pick a round booster and tech tile.

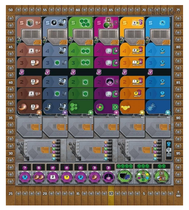


MapBoard

2. Before starting the first round of the game, every player can build two free mines on planets on the MapBoard. Players first take turns in a clockwise order, then in a counterclockwise order.

FactionBoard

3. The game will last a total of six rounds. During each round, the players progress through four phases in order: Income, Gaia, Actions, Clean-up.



Tech & Store Board

4. At the beginning of each round, income allows each player to gain resources. The faction board, round booster, tech tiles, and current level in each research area (above) all allow players to gain income.

Scoring Board

5. On the top of each faction board, there are four areas (bowls) holding power tokens. Tokens in the Gaia area must be moved to the area I of the power cycle in the Gaia phase. The player cycles through the Power system, and when the final stage is reached, the contents are spendable.

Tech Tiles

6. Starting with the first player and continuing in clockwise order, players take turns taking a single action. This continues until all players have taken the "Pass" action, which they do when they run out of things to do.

Terraforming Steps

7. There are nine actions in total: building a mine (colonizing a planet), starting a Gaia project, upgrading existing structures, forming a federation, upgrading technology level, power and Q.I.C actions, special actions, free actions, and pass.

8. The Clean-up phase prepares you for the next round. At the end of round 6, skip this phase and proceed to score.

Team Members

Yifan Zhou:
Email: yzhou106@dons.usfca.edu

Zheng Li
Email: zli124@dons.usfca.edu