AI-BASED OPTIMIZATION OF FARMING STRATEGIES IN BLOONS TD BATTLES 2

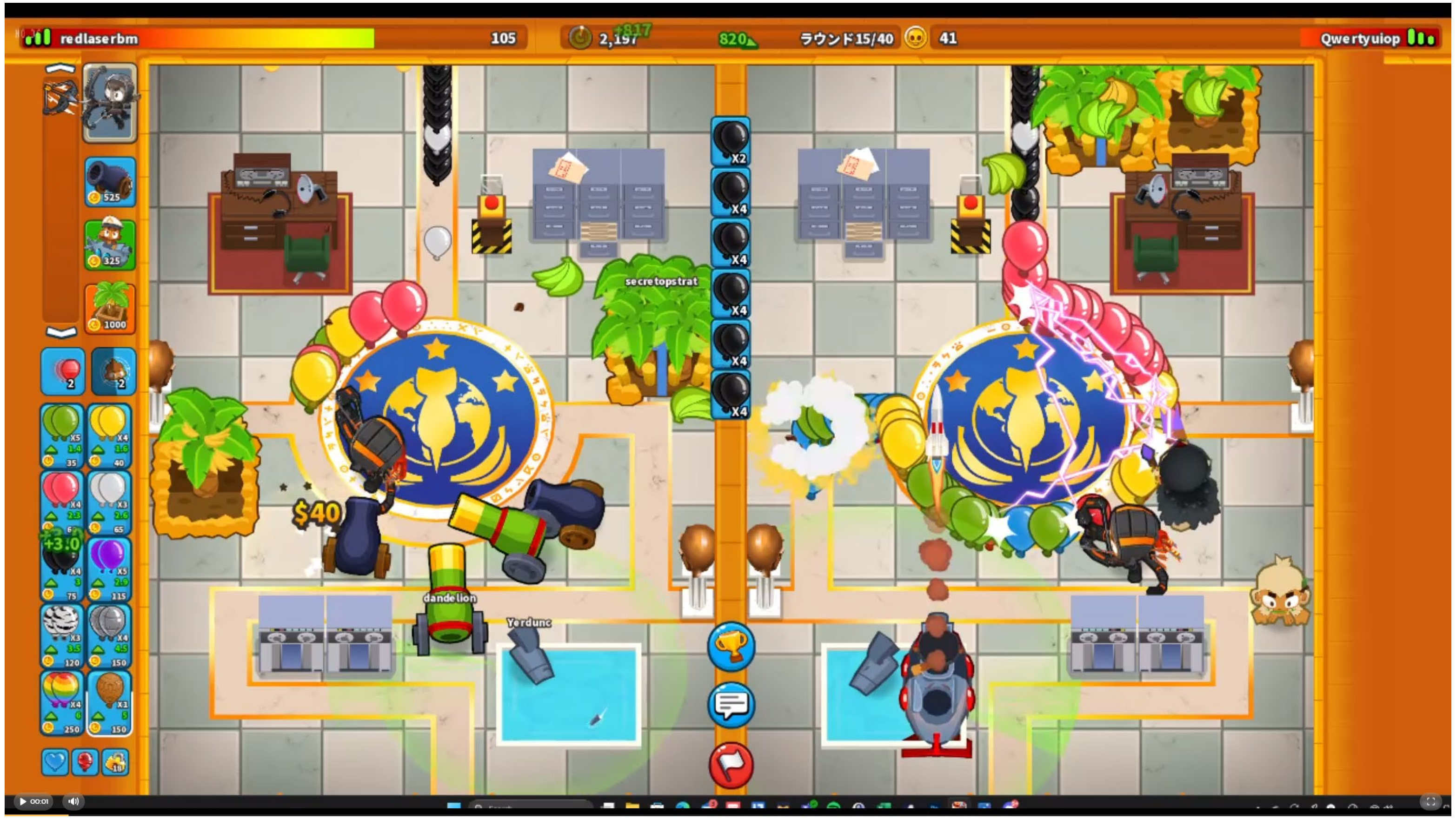
redlaserbm

OVERVIEW OF BLOONS TD BATTLES 2

Multiplayer versus game where players must send "bloons" to overwhelm their opponent while simultaneously popping bloons sent their way.

A game usually consists of up to 40 rounds. Each player has a limited selection of bloons they can choose to send, and this selection becomes increasingly more powerful as the game proceeds.

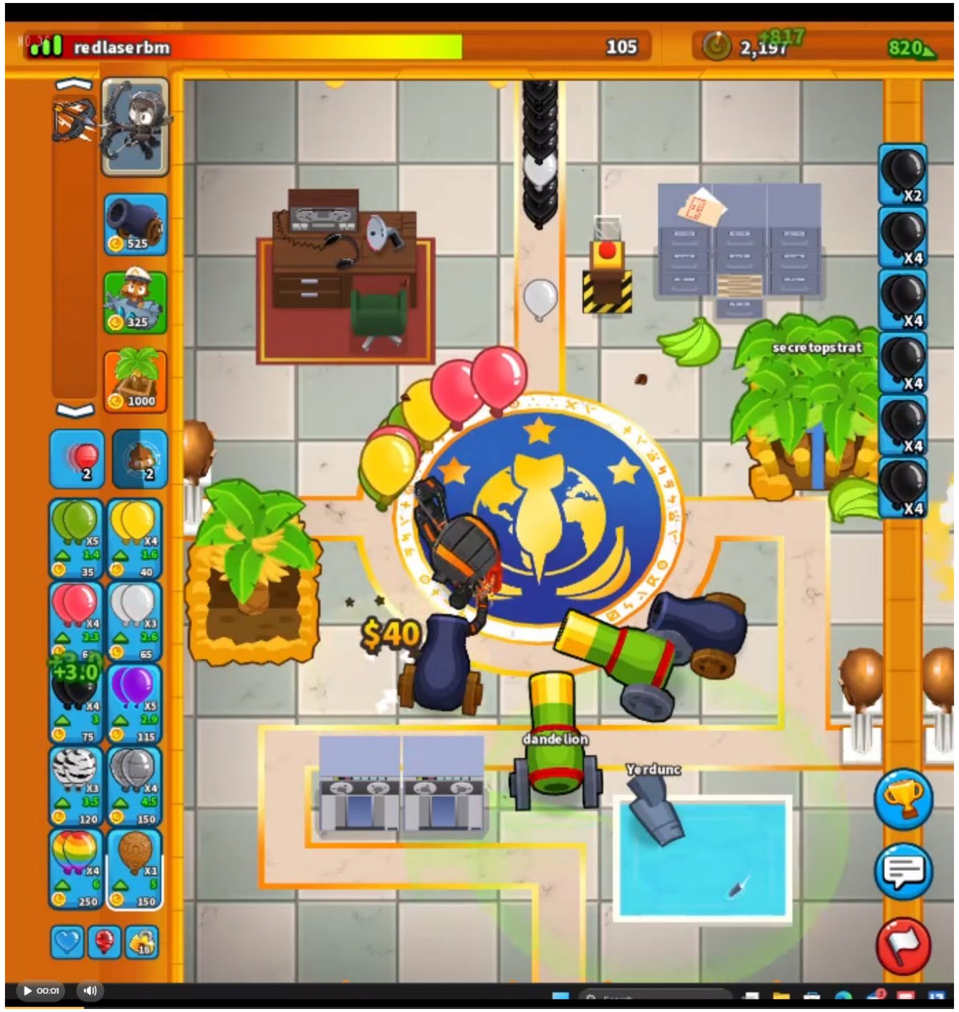
Optimal gameplay usually comprises building income early sending weaker bloons to sustain sending a lot of more costly, more lethal bloons later in the game.



GENERATING INCOME

Two primary sources of income:

- **Eco** – Cash gained every 6 seconds. The player can send bloons which cost money now but increase this value, allowing for more money later. Usually a cost-effective source of income generation.
- **Farms** – A source of income that pays out money evenly over the course of a round and can be upgraded as the game progresses to generate more money. The effectiveness of farms vary depending on round lengths.
- Top players normally focus on developing *both* income sources simultaneously early on before stopping eco at around R20 to develop the powerful Monkey Wall Street farm by the end of R24

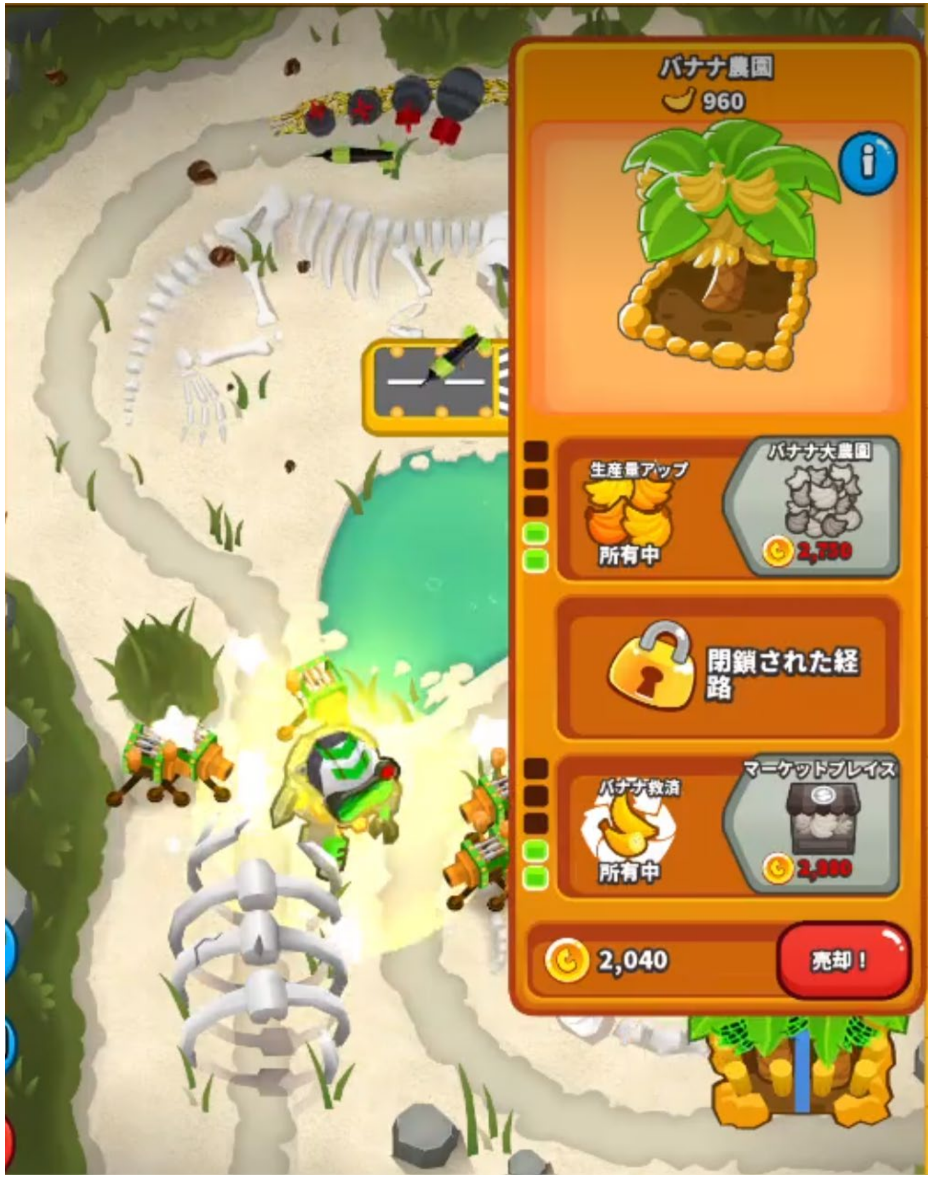


GENERATING INCOME - FARMS

Farms can be upgraded into one of three different paths subject to the following rules:

- Only two paths on a given farm may be upgraded at a time
- Only *one* path can be upgraded 3 more times
- If a farm is upgraded 5 times on a given path, no other farm may *also* be upgraded 5 times on that same path

We use the notation (x,y,z) to describe how many times a given path on a farm has been upgraded.



Above: The game UI shows a 202 farm. It is no longer possible to upgrade the middle path. If the top/bottom path is upgraded, it will no longer be possible to upgrade the bottom/top path.

PROBLEM OVERVIEW

Problem – It is difficult to rigorously determine how to best balance developing the two sources of income since their effectiveness can vary from game to game. Most top players go by “feel” which makes it difficult to identify the better of any two roughly similar farming flowcharts.

Stakeholders – Top-level players of Bloons TD Battles 2, who desire a way to objectively determine the best way to build income over the course of a typical game.

Task – Given a current *game state* (determined by the current game time, cash, eco, and farm configuration held) and a predetermined target time, determine the best way to build income from now until the target time.

NEAT ALGORITHM

A collection of neural networks, or “genomes”, are initialized with random weights and asked to perform a task at hand.

At the end of the task, each genome is given a fitness score representing how they performed on the task. Genomes with similar structures are grouped into species, and the best performers within each species are asked to “reproduce”, a process by a which a new genome is created using common traits from the parent genomes.

Genomes may also *mutate* during this process, opening the door for the network structure to become more complicated as deemed necessary.

Each iteration of running the task and reproducing constitutes a “generation”. After some number of “generations”, the process is terminated and the genome with the highest fitness is declared the winner.

METHODOLOGY DESCRIPTION

Inputs are (cash, eco, time, farm income)

Outputs are (eco intensity, farm intensity)

- Eco intensity varies from 0 to 1 and influences how aggressively the AI pursues building eco. 0 represents sending no bloons while 1 represents sending grouped purples.
- Farm intensity varies from 0 to 1 and influences how strongly the AI factors cost into its decisions on what farms to pursue next.

The AI is polled every quarter second on the action to take next. Fitness is evaluated based on the following:

- After reaching a certain round, the AI computes the amount of money generated over the next 4 rounds afterwards. This value is used as fitness.
- To avoid reaching outcomes atypical of optimal play, the AI is penalized for outcomes which fail to achieve a certain amount of eco and penalized for rapidly changing eco sends.

The AI is trained on the python b2sim library which I developed in June 2023, which allows for a substantially more time efficient way of seeing out the outcome of a given farm strategy than asking the AI to play the game outright.

The *initial* network structure that NEAT begins with. Hidden layer nodes may be added as training progresses.

EXPERIMENT DESIGN & RESULT

We asked the AI to optimize between the end of Round 13 and the end of Round 19, starting off with 0 cash, 600 eco, and two banana plantations crosspath'ed to have valuable bananas (which generates 1600 cash per round).

Typically, top level players in this scenario will choose to eco with grouped blacks until the start of Round 19, and then save up to upgrade one of the banana plantations into a banana research facility (the new farm configuration then earns 3800 per round).

AI Fitness	Human Fitness
52488	57360

AI performs well but misses the mark somewhat relative to human performance.

DESIGN FLAWS & FUTURE IMPROVEMENTS

Current infrastructure drastically simplifies the farming structure:

- AI only makes decisions based on farming income rather than the collection of farms on screen as a whole.
- AI influences farm purchase decisions by modifying the parameters of a utility rather than directly buying farms.
- This leads to faster learning of the concept of eco/farm tradeoffs but renders the AI unable to learn the finer points of optimal farming

Network performance is much worse if additional complexity is introduced:

- AI struggles to learn if given more advanced info about farms, tending towards clearly suboptimal strategies.
- Possible Solution: Perform supervised learning based on player data and then allow AI to evolve further in an unsupervised environment. To this extent, we could implement a network structure with an embedding layer to allow the AI to see and interpret for itself different farm builds and what they should lead to.

REFERENCES

K. O. Stanley and R. Miikkulainen, "Efficient evolution of neural network topologies," Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600), Honolulu, HI, USA, 2002, pp. 1757-1762 vol.2, doi: 10.1109/CEC.2002.1004508.