Ticket To Ride Board Game Playtesting Simulator

Team 37: Rachel Bruflodt, Patrick Cowhill, Jessica Houseknecht

Git Repository: https://github.gatech.edu/rbruflodt3/TTRSimulation

Table of Contents

Abstract

There were two main objectives of this simulation study: 1) to build a realistic virtual simulator of the board game "Ticket to Ride" that matches the real game as closely as possible, and 2) to evaluate the impact that different player strategies and rule modifications can have on the "fun" aspect or enjoyability of the board game. The game is simulated using an agent-based system coded in MATLAB, and game metrics are collected and analyzed to measure the effects that various game parameters have on game enjoyability. The study also considers the effects of different player interactions on the outcome of the game experience. By understanding the effects of various player strategies, rules, and interactions, the information gathered shall be useful in understanding and tuning the critical parameters and settings that are significant in creating a fun "Ticket to Ride" game.

Project Description

Playtesting a board game refers to a process by which a newly designed game is played by real-life players in order to evaluate the game in real-world conditions. Playtesting is a crucial part of board game design and provides a way for game designers to discover flaws or problems with a game before it is released. It also allows game designers to evaluate how well a game meets different design goals such as balance, play time, or strategic complexity. Playtesting, however, can be very time-consuming, especially for games which include randomness elements that necessitate playing the game many times to ensure coverage of many possible random states. Additionally, while playtesting can easily illuminate certain design flaws such as glaring balance issues, the ultimate question of whether the game is "fun" can be difficult to assess confidently with only a small number of playtesting sessions.

For this project, we have implemented a simulation of the popular board game Ticket to Ride (TTR) which can be used to playtest the game without real-life players. The simulation instead uses heuristic-based player agents intended to emulate strategies which real-life players might employ. The created player agents are not intended to find or model an "optimal" strategy for winning the game, but rather are intended to represent variations in how real-life players may choose to play.

The goal of this project is to make it possible to thoroughly playtest different iterations of TTR at the push of a button. We will demonstrate how our simulation can be used to quickly and easily assess the impact of rules and player variations on the experience of playing the game. In particular, we will analyze the impact of variations on the "fun" aspect or enjoyability of the game.

Literature Review

To tie the simulation goals in with measurements of "fun," a wide knowledgebase was surveyed around aspects of gameplay and user experience to establish a strong baseline definition of what "fun" looks like. In fact, there are accepted measures of gameplay that are both subjective (qualitative) and objective (quantitative) measures of game enjoyability. Measuring these aspects in a computer simulation and how to understand the implications of key metrics on enjoyment were also explored. Since the purpose of this study is to understand what *could* happen using computer simulation, the simulation shall attempt to measure user enjoyment and overall game "fun" quantitatively. The literature around game simulations was explored and gave insight into the types of game implementations and algorithms that have already been utilized to simulate gameplay. Additional focus was put into understanding the impact of player interactions on the enjoyability of a game.

Several criteria have been used to measure the user experience and/or "fun" nature of games. Studies have utilized surveys such as the GEQ as a subjective measure of user experience and enjoyability (Barbara, 2014; Barbara, 2017; Nutz, 2021). The survey consists of seven categories: immersion, tension, competence, flow, negative affect, positive affect, and challenge (IJsselsteijn, de Kort, & Poels, 2013). The user experience is measured on a Likert scale across all categories, and the GEQ has been shown to be effective for evaluating user experience in both computer as well as board games (Barbara, 2017).

Prensky (2001) also gives a description of important characteristics of an engaging digital game. He writes that "[g]ood game design" is balanced, creative, and focused (p. 23). It has "character," "tension," and "energy" (p. 23). He also emphasizes the importance of flow: not making the game too easy or too hard. There is cross-over on the list given by Prensky (2001) and the list on the GEQ used by Barbara (2014) and Barbara (2017): both highlight balance and tension and emphasize flow. Clearly, some of these characteristics of game design are subjective and will be challenging to measure. Some of the characteristics may be measured by proxy in this simulation study using some analogous metrics that can be modeled computationally. These are discussed later in the paper, but it is important to be aware of what game simulations have been performed in the present literature. In addition, subjective measures of fun may be difficult to validate without experimental data, so the study at hand shall focus on metrics that can be expressed quantitatively.

For this project we will implement an agent-based simulation of "Ticket to Ride" (TTR) very similar to the simulation created by Silva (2017). This paper outlines an approach for playtesting TTR using heuristic-based agents. In the paper, the authors explain the difficulty of implementing a typical search-based AI to play TTR due to the large number of possible game states and the stochasticity embedded in the shuffled decks of cards. The authors also argue that for the goal of playtesting, it is actually not desirable to use an optimized AI player, as the purpose is to analyze how typical human players interact with the game and its rules. For these reasons, the authors elected to create six different player agents with distinct heuristic-based strategies in order to simulate the game. They, then, analyzed the results of their simulation with a focus on the effectiveness of each strategy and the "most desirable" cities and routes for each game board (p. 1). Our intent for this project is to extend this paper by further exploring this kind of simulation's utility as a playtesting framework. We intend to use our simulation to explore how variations of the basic game may achieve different design goals related to the "enjoyability" of the game.

Holmgard (2014) presents an approach for defining heuristic-based strategies intended to emulate human players of a video game. The player agents in the paper attempt to model human decision-making using the concept of "utility," which the authors describe as "how much an expected outcome of a decision is worth to the decision maker versus the expected cost and risk of attaining that outcome" (p. 2). First, the game mechanics are analyzed to determine sources of utility, and then player agents are defined by positively or negatively weighing these utility sources. The player agents can be evaluated by how much they maximize utility with their given strategy.

Before examining the standard methodology of performing statistical analysis on measurable parameters of a board game simulation, an important distinction to make is what quantitatively measurable features of a system and actors within the system are important to the overall goal and conclusion of the simulation. In this study, as artificial players compete to win countless rounds of "Ticket to Ride," the goal of maximizing these synthetic actor's "fun" must be broken into understandable, separable parameters, as "fun" itself cannot be represented solely scalar and measured directly. Multiple prior examinations into similar veins of research reveal potential avenues for describing fun as a collection of finite, measurable characteristics.

Many of the resources described earlier in this review analyze the impact characteristics of the game can intrinsically have on the production of "fun" for all players, independent of their own disposition. These characteristics include the balance of the game, the goals and objectives, and the artistic representation and story accompanying the interactable experience (Prensky 2001). However, as important as the intrinsic board game quality is, the personal experience of the player or group of players participating in the game is another crucial factor to consider. Prensky (2001) considers factors such as interactions between players (and between players and the game), inter-group competition, player-to-player opposition, and player growth to be just as important when evaluating the enjoyment a game can provide to its players. These factors can be divided into two primary interactions: (1) those resulting from the players' relationship with the boardgame, the gameplay, the pieces, the art, and other non-human elements of play and (2) those resulting from the players' relationship with one another, teamwork, disagreements, shared reward, contrasting strategies, and other social dynamics present in the guided human interaction a board game is responsible for.

Chanel (2011) explores the first category of these quantitative "fun" factors, primarily by relating skill of player and difficulty of game to overall enjoyment. Chanel (2011) bases the optimal relationship between independent features of the player and features of the game within the "flow channel," which is simply referred to as "flow" by Prensky (2001), Barbara (2014), and Barbara (2017). They describe a positive correlation between skill and difficulty as residing in this "flow channel." This results from low-skill players playing high-difficulty games producing a response of anxiety and high-skill players playing a low-difficulty game producing a response of boredom. The Goldilocks channel that runs from low skill and low difficulty to high skill and high difficulty is where the most enjoyment should be found. Chanel (2011) importantly relays this to the difficulty curves found in many computer games. As the player becomes more skilled at playing the game, the game should (either statically or dynamically) increase in difficulty to maintain the focus and attention of the player. In this study, players can be simulated anywhere on the spectrum between beginner and expert, and to maximize fun, the synthetic version of "Ticket to Ride" that is being tested should provide avenues for players of all skill levels to exercise their experience to play the game in diverse ways. Additionally, in contrast to Chanel's (2011) study where the single-player game of Tetris was being examined, this study describes a game with multiple players. That means to minimize the deviation from the "flow channel" for every player playing the game, the difficulty available for players of different skill levels should reflect their experience and abilities in the game. As competition is a significant feature of "Ticket to Ride," maximizing "flow channel" exposure is a difficult feat and will be heavily influenced by the design of the synthetic version of the game.

With the consideration that many games played are only as fun as the group playing them, Xu (2011) explores the impact and importance of social play in board games when designing an enjoyable experience. Xu's (2011) study investigated the effects of separating social elements in four different board/card games by using computers to limit social interaction between players. In doing this, they found that certain types of social interaction deriving from gameplay can be linked to specific categories of events that occur in board games, and much of the enjoyment gained or lost from playing the game could be attributed to the outcomes of these social interactions. For example, they found that games that pit players against each other and reward players who intentionally and directly sabotage the success of their peers were more enjoyable by a group who all enjoyed that type of joking, pranking, and making-fun style of progression. In the case that a single person in the group did not appreciate that singling-out social dynamic, their enjoyment of the game was negatively affected by the behavior of the other players. Likewise, a more isolated or cooperative game, in which there are no means of bringing down other players, attracted a group consisting of players who enjoy fair play and steady progression where the best player (and not necessarily the most cunning and devious) wins. Xu (2011) found that even individual elements of a game or events that occurred during a game can have micro-rewards and micro-punishments to some or all the players in a group by way of an "interaction ritual" (p. 4). This "interaction ritual" is described as the way that game actions direct the relationships and interactions between players and can be a factor in building social engagement in a game. While certain events, outcomes, actions, or processes produced by the game can have this effect, even stagnant, static elements of a game can guide social interactions between players. Xu (2011) provides the observation that players participating in-person tended to maneuver and gesture with their pieces and tokens when discussing strategies with one another. This served as a tactile element of the imaginary story being told throughout the game. When examining the effect of a synthetic version of "Ticket to Ride" on the players' enjoyment, coupled interactions within the group and between players because of dynamic actions and static elements of the game should likewise be considered an important dimension to explore during this study.

To tie everything together, specific metrics for game simulations shall be discussed. As stated earlier, one of the key aspects to good game design is balance. Jaffe, Miller, Andersen, Liu, Karlin, and Popovic (2012) describe several measures of balance in a game. They define "game balancing" to be the "fine-tuning phase in which a functioning game is adjusted to be deep, fair, and interesting" and state that it requires designers to "repeatedly tweak parameters" and usually conduct extensive playtesting "to evaluate the effects of these changes" (p. 1). They attempt to design a framework for rapidly evaluating the balance in games. They offer several questions that one might consider when evaluating how balance is relevant to their game, as balance will look different depending on the type of game.

To address these questions in their card game simulation, they established quantitative restrictions, which acted as balance measures, in gameplay. This allowed them to assess items such as the importance of a particular action or whether a certain card was "overpowered" or "underpowered," leading them to adjust rules or parameters accordingly (p. 29). The authors formulated a few quantitative restrictions to answer balance questions, two of which are shown in the table below.

Two Balance Questions and Corresponding Balance Measures

Balance Question	Balance Measure and Model
How powerful is a given action or combination of actions?	'Omit a': $\lvert Plays(a)\rvert \le 0$. 'Prefer a': $\lvert Plays(a)\rvert \ge \infty$.
How much long-term strategy is necessary?	'Random': Depth ≤ 0. 'Greedy': Depth ≤ 1.

Note. Taken and reformatted into a table from the balance questions and measures given in "Evaluating Competitive Game Balance with Restricted Play," by A. Jaffe, A. Miller, E. Andersen, Y. Liu, A. Karlin, and Z. Popovic, 2012, pp. 27-28.

In essence, the authors found analogous metrics to balance and created formulations that can be implemented computationally and measured mathematically. They then compare the win rates of the players under the given restrictions or balance measures. The power behind taking certain actions during a game under specific parameters could be implied by the distributions of the player win rates, and the relationships with scores with several final game metrics

The present study, likewise, creates metrics for the "Ticket to Ride" simulation to evaluate the game power and balance. The power behind taking certain actions during a game under specific parameters could be quantified by understanding the distributions of the player win rates, and key relationships between final scores and other output results. If the game result is known long ahead of time (in which case, long-term strategy might also be lacking), then this could negatively impact game experience. The current paper expands on this by keeping track of the points scored after each turn of the game with and without long route points. The TTR Simulation built has a "Dummy Player" who uses a simple greedy strategy, which generally performs worse than other players that execute more complex, long-term strategies, indicating that long-term strategy is important in the game. Finally, the win rates of all players (which should be close in a balanced game) and their statistical significance can help gauge game balance.

Holmberg and Modee (2021) speak to challenge, balance, and freedom as important aspects of gameplay. The results of this paper reveal that the non-random rewards system was more enjoyable and offered greater autonomy to the players. In the non-random system, players were allowed to choose their reward, whereas the random system assigned a random award. Similarly, this study explores whether we can create rules that offer more autonomy and "non-random rewards" to players by allowing them to choose from a set of actions, which may help their chances of winning (p. 1). The current study's simulation results have shown that a player can win the game without completing destination ticket cards. The team has considered and run simulations that add a non-random extra reward (+7 bonus points) for completing destination ticket cards, which may allow for greater balance, autonomy, and reward.

"Challenge" is an aspect that arises frequently in the literature around enjoyable gameplay. Abuhamdeh and Csikszentmihalyi (2011) show that relative chess rating (which may be a measure of challenge or difficulty) and enjoyment of chess players was a downward facing "U-shaped" curve with right skew (p. 318). That is, most players enjoyed games more when they were playing someone with a higher rating than them; then, up to a point "(i.e., the apex of the curve)…further increases in difficulty" led to "decreases in enjoyment" (p. 318). In addition, the results showed that "close" games were more enjoyable than "blowouts" (p. 321).

This suggests a couple different ways that this study can explore the "Ticket to Ride" simulation. First, it may be possible to compare the ability of players based on strategy by performing several iterations of the simulation and measuring the win rate. Guhe and Lascarides (2014) state that "win rate" measures "an agent's overall success" in their agent-based Settlers of Catan simulation (p. 3). Like the Settler of Catan study, the current study also uses a heuristically defined agent-based simulation strategy, where the agents represent individual players of the game. Guhe and Lascarides (2014) describe how they evaluate the effectiveness of a strategy based on whether the win percentage is statistically significant, which was also done in this study.

Given the enjoyability of "close" chess games mentioned in Abuhamdeh and Csikszentmihalyi (2011), the current study also measures the average difference in total points between each rank (e.g., 1st vs. 2nd, 2nd vs. 3rd, etc.) at the end of the "Ticket to Ride" game to understand the "closeness" of an average game at particular parameter settings. If one were to run several iterations and measure the difference between the two closest point totals at the end of the game, for optimized enjoyability, it will be important to have this average difference low (which would indicate there is a small gap between the winner and runner-up average scores) relative to points awarded in the game.

Conceptual Model

The Ticket to Ride Board Game

Setup and Objective

To understand the system being studied and establish a baseline against which to measure changes on game parameters, it is important to define the standard board game rules, pieces, and objectives. "Ticket to Ride" is a two to five player board game. It consists of a board containing colored tracks or "routes" between cities.

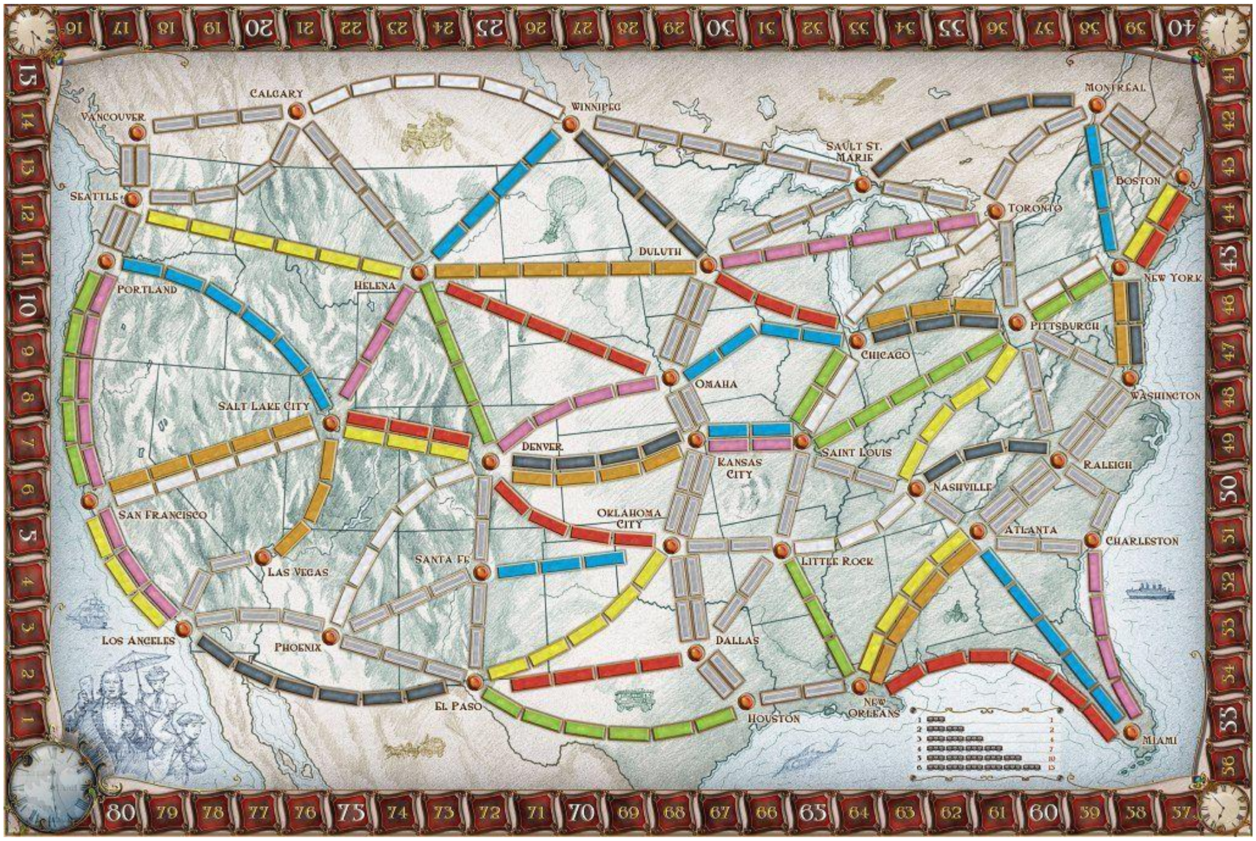


Image of the Ticket to Ride board showing the colored routes

The game consists of two types of cards: “Train Car” cards and “Destination Ticket” cards. At the beginning of the standard “Ticket to Ride” game, players are each given 45 colored train pieces, four colored train cards, and three destination ticket cards. Five of the “Train Car” cards are left face-up in a pile for all players to see, and the rest of the cards are face down in a draw pile. The objective of the game is to earn the most points by the end of the game, which can be accomplished by claiming routes between neighboring cities, completing “Destination Ticket” cards, and building the longest path.

The destination ticket cards each have a point value attached to them, and the player earns those points by connecting contiguous routes between the two locations specified on the destination ticket cards. At the beginning of gameplay, the player must choose at least two of the three destination ticket cards he or she picks.

Taking a Turn

On a player’s turn, the player can take one of three actions: claim a route, draw destination cards, or draw a card from the draw pile and face-up cards.

Claiming a Route: “Train Car” cards are used to claim routes between two cities. A route’s length is the number of spaces on the board between the cities. A route can have between one and six spaces, and to claim the route, the player must have train cards of the same color equal to the number of spaces in the route. Locomotive cards can be considered "multicolored" and used with any color route. When a player claims a route, he discards the appropriate number of "Train Car" cards and puts the same number of train pieces on the board, indicating he owns the route. His score is then increased by the point system given by the corresponding number of trains. For example, a 6-train route is worth 15 points.

Draw Destination Cards: In the middle of the game, if a player decides to pick more destination cards, he or she picks three from the deck (if available) and must choose to keep at least one. Any cards the player does not keep must be discarded.

Draw a Card from the Draw Pile or Face-Up Cards: A player has the choice between picking two cards from the draw pile or the face-up deck. If picking a card from the face-up deck, the player must immediately replace it with another card from the draw pile. If the player chooses to pick a multicolored locomotive card from the face-up cards, he or she may not choose another card from the face-up cards nor the draw pile.

Ending the Game

Play proceeds until one player has two or less train pieces remaining. Then, all other players get one more turn. Any destination cards completed during the game are awarded points at this time. However, any destinations that were not completed are subtracted from the player's total score. Finally, ten points are awarded to the player with the longest continuous route.

Player Agents

The role of the player agents in the simulation is to choose which action to take based on the available choices. In TTR, each player must choose:

- Which destination ticket cards to keep at the beginning of the game
- Which action to take on each of their turns

Several different types of player agents were modeled to reflect the various strategies of real-life players.

Utility-based Strategic Players

A generic decision-making model was created to model players choosing actions based on the "utility" of each action. Utility here refers to the value of taking the action as perceived by the player, or how useful the action is for implementing the player's chosen strategy. The decision-making model is defined as follows:

- Each route on the board has a numeric utility value calculated by the player which reflects the importance of claiming the route for the player's strategy. The route's utility may change over the course of the game depending on the player's strategy and the state of the board.

- Points in TTR can only be earned by claiming routes and routes are claimed by discarding train car cards. Therefore, train car cards have “potential” utility which reflects the value of drawing the card or having the card in the player’s hand.
- The utility of drawing or keeping train car cards is discounted by a “potential discount” calculated by the player. This discount is to reflect the fact that as the game goes on, it becomes less likely that the potential utility of a train car card will be realized, since the player may simply run out of turns or trains. For example, on the last turn of the game, it doesn’t make sense for a player to draw cards if they are able to claim a route, since the cards will never be used. In this case, the potential discount would set the utility of drawing cards to 0, so the player would take the more logical action of claiming routes to earn a few more points before the game ends.

On a utility-based player’s turn, they follow these steps:

1. For each unclaimed route on the board, calculate the utility of claiming that route. Details of this calculation are discussed later in this section.
2. The player calculates the potential utility of each color of train car card based on the previously calculated route utilities. The utility for each color is calculated as the utility of the highest-utility route of that color divided by the length of the route, since the route length indicates the number of train car cards needed to claim it. The utility for locomotive cards is set to be the same as the color with the highest utility, since the locomotive could be used to claim routes of that color. The utility for gray routes is handled in a special way, since any color could be used to claim a gray route. The gray color utility is first calculated the same way as the other colors, but then if any color has a utility less than the gray utility, that color’s utility is set equal to the gray utility. This is because cards of that color would have more utility being used for a gray route than a route of the specific color.
3. The player calculates their potential discount based on the current state of the game. For all currently implemented players, the potential discount is based on the minimum number of train pieces any player has yet to place on the board. This is because the game ends when any player has less than 3 train pieces left, so the number of train pieces is a good indicator of how close the game is to ending.
4. For each route that the player is currently able to claim, the player compares the calculated utility of claiming that route with the potential utility of the cards which would be discarded to claim the route. This step ensures that players do not just simply claim routes as soon as they are able to, but instead are able to save cards that are needed for higher priority routes.
5. If any of the claimable routes have a higher utility than the cards’ potential utility, the player claims the highest utility route for which this is true.

6. If the player has not chosen to claim any routes in the preceding steps, they compare the potential utility of each of the drawable train car cards and select the highest utility card to draw.

This decision-making model provides a generic framework for players who are able to prioritize routes and execute simple long-term strategies. Differentiation in player strategies is then achieved in the calculation of route utilities, as players will value routes differently based on their strategy. The route utility calculation is modeled as a weighted sum of three different types of utility: route length, destination ticket completion, and "deviance."

Long Route Player
The Long Route player is focused on claiming long routes regardless of where they appear on the board and is not at all concerned with completing destination ticket cards. The utility of claiming a route for a Long Route player is simply the points earned for claiming the route based on the route's length.

Destination Ticket Player
The Destination Ticket player is focused on completing as many destination ticket cards as possible. The utility of claiming a route for a Destination Ticket player is based on how useful the route is for completing any unfinished destination ticket cards in the player's hand. When calculating a route's utility for destination ticket completion, first the smallest number of train pieces needed to complete the ticket given the current state of the board is determined. Next, the smallest number of train pieces needed to complete the ticket if the player were to claim the route is determined. The utility is calculated as

$$\left(1 - \frac{\#\ with\ route\ claimed}{\#\ without\ route\ claimed}\right) * points\ the\ destination\ ticket\ is\ worth$$

Deviant Player
The goal of the deviant player is to sabotage the strategies of other players by blocking them from claiming long routes or completing destination ticket cards. The deviant player keeps track of the colors of train car cards that other players draw and attempts to guess which routes the other players may be trying to claim, then tries to claim them first. To calculate a route's utility for a deviant player, first a "color factor" is calculated for each opposing player which reflects how many cards of the route's color the player has been observed drawing. Next, a "connection utility" is calculated which reflects how well the route connects to the opposing player's other claimed routes. This calculation is similar to the destination ticket utility calculation and is meant to model the deviant player attempting to block other players from completing destination cards. After evaluating all opposing players, the maximum connection utility is multiplied by the player's color factor. Next, a "deviant length utility" is calculated by taking the maximum of the opposing players' color factors and multiplying by the route's Long Route utility. This is meant to model the deviant player attempting to

block opposing players from claiming long routes. The route utility for the deviant player is then the maximum of the "connection utility" and "deviant length utility."

Varying the weights of each type of utility in the route utility calculation provides a way to model players who attempt to implement multiple strategies to varying degrees.

Treacherous Player

Since the default ruleset for Ticket to Ride makes it difficult for a player to directly target other players with attempts to obstruct their progress, the Treacherous Player is a simulation agent specially designed with the unique ability to take advantage of additional rules and actions that enable them to prevent the claiming of routes and completion of destination tickets more successfully. The Treacherous player is able to perform any of the following non-standard actions during a game using the Treachery Ruleset:

1. <u>Train sacrificing</u>: Once on a player's turn, as a free action (in addition to the action they take), they may choose to permanently discard one of their un-played rain pieces to draw an additional train card from the draw pile.
2. <u>Route Stealing</u>: If a desired route is already owned by another player, a player has the option to spend an additional two trains to "steal" the route for themselves. The trains of the player who previously owned the route would not be returned to the player and would instead be discarded from the current game. The stealing player would place their trains on the route, and all the benefits of claiming the route such as gaining the points for claiming the route and points for completing any destination tickets would be granted to the stealing player as if they had claimed the route.
3. <u>Route Blocking</u>: A player may discard half the necessary number of cards to claim a route (rounded up) as well as a number of train pieces equal to the length of the route to render a route unusable. The player does not gain any points that they would if they claimed the route, and the route is not considered when connecting cities for destination tickets or calculating the longest route at the end of the game.

Dummy Player

The Dummy Player employs a simple "greedy" strategy. On each turn

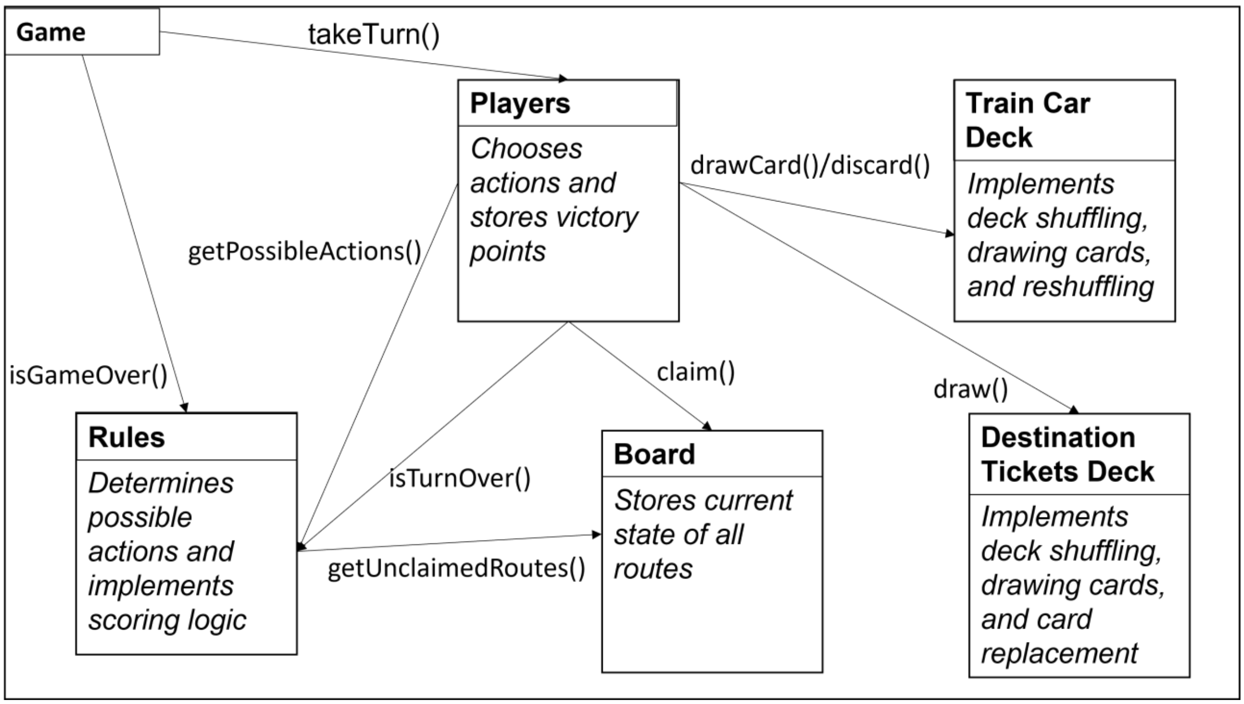
- If the player is able to claim any route, they claim the longest route they are able to claim
- If the player is not able to claim any route, they draw a card from the top of the train cards deck

The Dummy Player was used for testing the simulator before more complicated strategic player agents were implemented and also serves as a kind of "control" player for experiments.

Simulation Model

Overview

The simulator was implemented in MATLAB as an object-oriented program using MATLAB classes and enumerations. The top-level class is the Game class which includes all of the logic and components necessary to simulate a full game of TTR. The Game class is composed of a Board object, a Train Card Deck object, a Destination Card Deck object, a Rules object, and an array of 2-4 Player objects. The following diagram shows the major interactions between the objects during a simulated game.



To simulate a full game, the Game object iterates over the list of players, calling the takeTurn() function for each. After each turn, the Game object updates the Rules object with the current state of the game, and the Rules object determines whether or not the game has ended. Each of the class components of the Game class are extendible and variant simulations can easily be created by simply initializing the Game object with different combinations of boards, rules, decks, or players.

Game Component Modeling

Board

The Board class contains a representation of the current state of the board modeled as an undirected graph. The edges of the graph are the routes on the board and contain information such as route length and which player currently owns the route. We utilize the built-in MATLAB graph class, which allows for easy analysis and path calculations.

Train Car Deck

The TrainsDeck class keeps track of the state of the train card deck. The draw pile, face-up cards, and discard pile are represented as arrays of TrainCard objects. The TrainsDeck class handles the logic for drawing cards, replacing cards in the face-up pile, and reshuffling the discard pile when needed.

Destination Card Deck

The DestinationsDeck class models the destination card deck as an array of DestinationTicketCard objects. The class include methods for drawing cards or returning cards to the bottom of the deck.

Rules Modeling

The abstract Rules class is the top-level parent class for all rules variations. A Rules object is responsible for:

- Determining which actions a player is allowed to take based on the current state of the game and any other actions the player has already taken this turn
- Determining when a player's turn is over
- Determining the point value awarded for claiming a route
- Determining when a game is over
- Updating player scores at the end of the game to include destination ticket awards and longest route bonuses

These responsibilities are declared as abstract methods in the Rules class and must be implemented by any child Rules class. With this implementation, all of the logic for determining what a player is allowed to do given the current game state and all of the scoring logic is contained entirely within the Rules object. This makes it easier to test rules variations, as one can simply implement a new Rules child class without changing any other components.

Functions for determining a player's longest route at the end of the game and determining if a player has completed their destination cards are also implemented as static methods in the base Rules class.

Player Modeling

The abstract Player class is the top-level parent class for all player agent classes. As stated before, a Player object is responsible for:

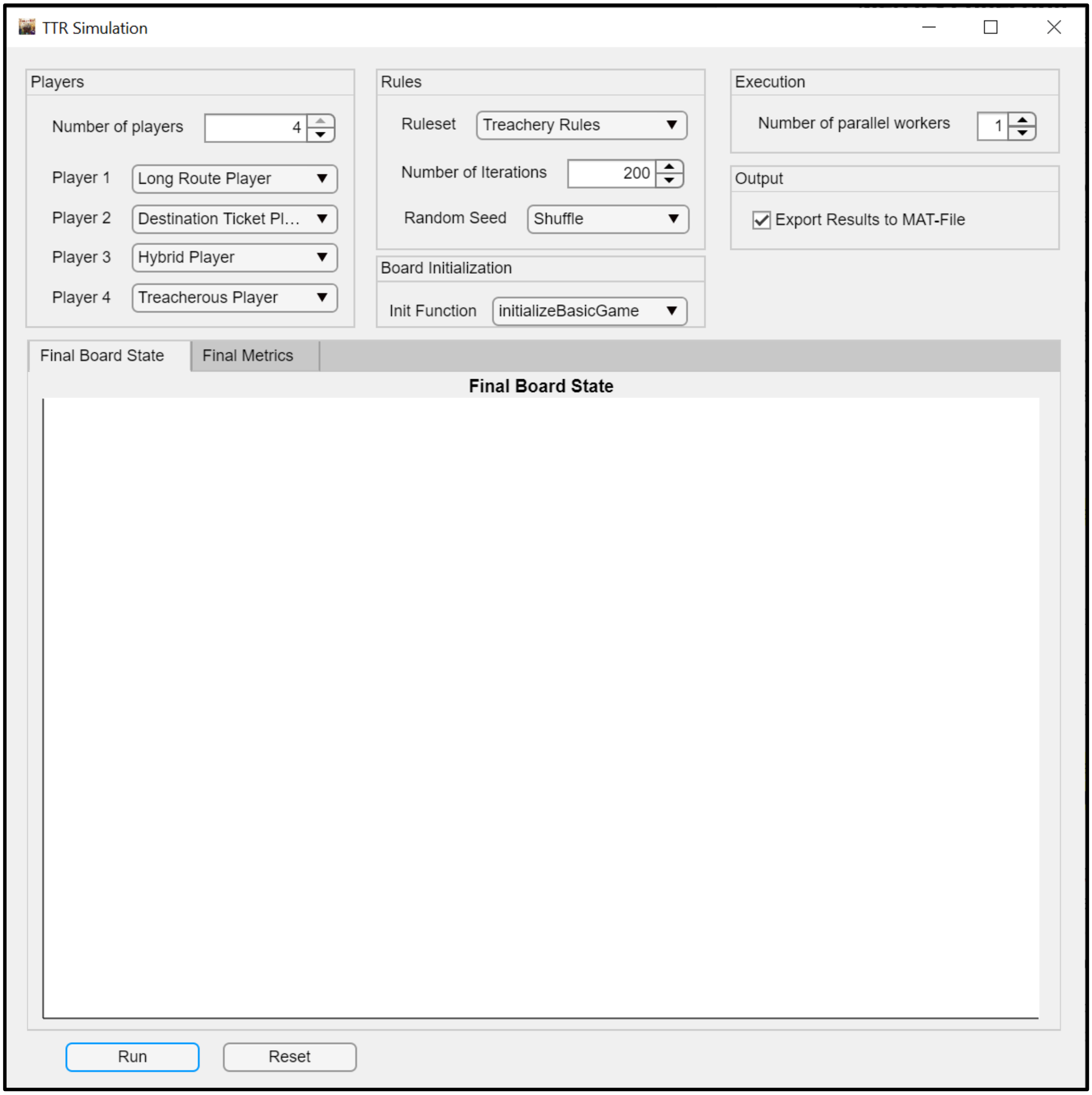
- Choosing which destination ticket cards to keep at the beginning of the game
- Choosing which action to take on each of their turns

Similar to the Rules class, these responsibilities are declared as abstract methods in the Player class and must be implement by any child Player class. This means that any implemented player agent need not be concerned with determining what actions they are allowed to perform, as they simply choose from a list of possible actions using any suitable method.

The top-level Player class contains the takeTurn method which is called by the Game class. When takeTurn is called, the Player class calls the given Rules object's method to determine the possible actions the player may take. The possible actions are returned as a struct. The Player class then calls the abstract method for choosing an action, which is implemented by a child class, passing the struct of possible actions as an argument. The child class's method must return one of the possible actions provided in the struct. Once chosen, the Player class carries out the child class's chosen action, discarding and drawing cards as needed. With this implementation, the only necessary functionality of the child Player classes is choosing actions, since the implementation for carrying out any action is contained within the top-level Player class. This makes it easier to implement many different types of player agents and reuse logic as much as possible.

Simulation Interface

A MATLAB app was designed to allow users to interact directly with the simulation by both setting up games through a selection of player strategies, rulesets, parallelization options, etc. and receiving comprehensible feedback in the form of a visual representation of the final state of the game board, a log of actions taken by players throughout the game, and charts displaying numerous metrics that describe the results. Producing and maintaining this UI throughout this study has consistently provided quick, easy, and well-defined access to a suite of adjustable parameters and has enabled the accelerated discovery of interesting results during experimentation.



Metrics Collection

Game Metrics Output Structure

Several metrics were reported during each simulation. This simulation was a Bounded Horizon study. We ran a pre-set number of iterations of the simulation to be processed by a set amount of parallel workers or processors. All the simulation parameters were set ahead of time. Thus, it made sense for the simulation output to include confidence intervals around averages of key game metrics. The metrics were used to measure the "fun" aspect or enjoyability of a game and verify and validate the simulation.

Results Struct Output by the Run Simulation Function (Raw Data from Games)

When the program was simulating a game, individual metrics were collected during and after the game and stored in a game results struct. Individual player metrics from each game iteration were stored in a summary in the game results struct and the winning routes were stored in a separate array in the struct as shown in *Figure 1*. Note that the number of rows in the struct will depend on the number of iterations performed in the simulation. The figure below shows results from 30 games between a Long Route and Hybrid player 3.

Structure of the results struct formulated after 30 iterations of the game simulation

```
results =

  30×1 struct array with fields:

    summary
    winningRoutesTbl
```

Results Struct Output by the Process Simulation Results Function (Results for Overall Simulation)

After all games (iterations) were run, the results of all games (the entire simulation) were reported to the user at the command line as well as displayed in the bar charts, correlation plots, and final board printed out by the last game. From the UI, the user also has the capability of saving the results output from the simulation to a .MAT file for later comparison with other simulations. The following figure is an example of the structure of the struct output by the process simulation results function for a 4-player game (1 Long Route Player and 3 Destination Ticket players) with 100 iterations. The sizes of the struct fields will vary depending on the number of players and iterations performed.

ProcessSimulationResults output struct

```
struct with fields:

      combinedSimResults: [100×44 double]
        winningRoutesTbl: {100×3 cell}
          playerStatsTbl: [4×20 table]
        playerConfIntTbl: [4×30 table]
                winRates: [4×1 table]
   winRatesStatResultsTbl: [6×5 table]
                 corrTbl: [45×8 table]
           avgDiffPerRank: [3×2 table]
          playerRankings: [4×4 table]
                settings: [1×1 struct]
```

The following section will discuss the specific metrics output by the ProcessSimulationResults function by walking through each of the fields in the results struct described. Metrics were used for specific purposes such as analyzing balance and other characteristics that make a game fun, verifying and validating the simulation, and exploring the relationships between different final game simulation results metrics.

Player Stats and Confidence Interval Tables (playerStatsTbl, playerConfIntTbl)
This table contained 10 different metrics measured for each individual player in a game. The average, standard deviation, and 95% confidence interval (where applicable) were reported for each of the 10 metrics.

Number of Train Cards Left
This is a measure of the number of train cards a player has left in his hand at the end of a game.

Number of Destination Ticket Cards Completed
This is a measure of the number of destination ticket cards a player has completed (by connecting contiguous paths between two locations on the card) at the end of a game.

Number of Turns Taken
This is a measure of the number of turns a player took during a game. Note that for any single game, the number of turns taken will only differ by up to one card. This is the case because each player gets one more turn, once a player's total trains is two or less. This metric could be used to gauge how long games at certain settings typically take as well, whether there was a player who consistently had fewer turns by running out of trains first.

Number of Routes Claimed
This is a measure of the number of routes or legs between two locations that a player has claimed throughout a game.

Average Route Length
This is a measure of the average length of a route claimed by a player during a game. Note that longer routes earn more points for the player.

Length of Longest Route
This metric measures the length of the longest route claimed by a player during a game.

Number of Turns Ahead in Game
This metric measures the turns during a game in which a player was in the lead. In other words, after comparing current scores with the other players, this metric was incremented for the player(s) with the highest score(s).

Number of Turns Ahead in Game with Long Route Points

Extending upon the previous metric, this measures the turns during a game in which a player was in the lead, considering the longest route currently on the board. In the actual game, the long route points get counted at the end of the game, so the points are not officially finalized until the end. However, including the long route points in the current score may be an indicator of whether or not the outcome can be predicted early in the game.

Win Rates Table and Pairwise Significance (winRates, winRatesStatResultsTbl)

This metric was included in order to more directly compare the individual performance with different simulation settings. A pairwise t-test was performed to determine whether the win rate of a particular player (strategy) was statistically higher than that of another player (strategy). The MATLAB Statistics and Machine Learning toolbox were utilized to exploit the functions for acquiring the critical values.

Correlation Table for Player Stats (corrTbl)

A correlation table was created to explore relationships between metrics reported in the playerStatsTbl and confIntTbl. The data points in the correlation plot represent individual player values for each of the metrics. In other words, the correlations may be used to explore the interactions of very specific playing strategies and the kind of behavior and statistical relationships they produce. It can be a measure of the “power” and “balance” in gameplay with players of specific strategies. The correlation table measures include the correlation coefficient R, correlation coefficient confidence interval, the p-value of the correlation coefficient, R^2, and the confidence interval for R^2.

Average Difference in Points/Rank (avgDiffPerRank)

The avgDiffPerRank table was intended to measure the “closeness” of games. The first table measures the average difference in final score per rank (e.g., the difference in points between the winner and runner-up). This metric was intended to spot-check to ensure the given simulation settings did not cause any major blowouts (indicated by a large difference in score between ranking first and second place, for example).

Number of Times First and Last (playerRankings)

The playerRankings table was also intended to measure “closeness” of games. Player rankings contained information about the number of wins and losses of each player, the idea being that players who always lost would likely not have as much fun playing the game.

Winning Routes Table (winningRoutesTbl)

The winning routes table reports the number of times each route was claimed by the winner of a game. This may be used to measure the “power” and “balance” of game by potentially giving insight into which routes are most desirable or which destination ticket cards are most advantageous to draw. Another study on the Ticket to Ride simulation also looked at which routes were most often claimed by the winner (Silva, 2017).

Compare Results Functionality

After running simulations of various parameters, the user could load the results from individual games into the CompareResults MATLAB function to compare results of separate simulations set at different parameter settings. The CompareResults function then produces an output struct summarizing the compiled and compared simulation results from the different simulations.

Structure of the CompareResults Struct Explained

Following is the structure of the CompareStruct. Note: the raw data is saved in a .file field in the Compare Results struct. The bolded fields contain aggregate tables that contain the combined results of all the different simulations.

Results struct from ProcessSimulationResults.m

Level 1 and Level 2 Fields
.combinedSimResults
.winningRoutesTbl
.playerStatsTbl
.playerConfIntTbl
.winRates
.winRatesStatResultsTbl
.corrTbl
.avgDiffPerRank
.playerRankings
.settings (struct containing 7 Level 2 Fields)
  .nPlayers
  .nIterations
  .rngSeed
  .nWorkers
  .playerTypes
  .ruleSet

Results struct from CompareResults.m: Level 1 Fields
.fileID .file **.settings** **.playerStatsTbl** **.playerConfIntTbl** **.winRates** **.winRatesStatResultsTbl** **.avgDiffPerRank** **.playerRankings** **.corrTbl** **.winningRoutesTbl**

* **Note:** the **bolded** fields above indicate the new tables with the combined files' results. They are named similarly to the fields from the ProcessSimulationResults struct.

The following two tables display the structure of a CompareResults struct. The bolded fields below indicate that the field contains another level of fields in the structure. This particular struct models aggregate results from a call to CompareResults with only 2- and 4- player results files being compared. Any results stored in "nPlayers3" fields would be empty.

Results struct from CompareResults.m: Level 1-2 Fields

- .fileID
- .file
- .settings
- **.playerStatsTbl**
  - **.runData**
  - **.summary**
- **.playerConfIntTbl**
  - **.runData**
  - **.summary**
- **.winRates**
  - **.runData**
  - **.deviations**
- **.winRatesStatResultsTbl**
  - .nPlayersAll
  - .nPlayers2
  - .nPlayers4
- **.avgDiffPerRank**
  - **.runData**
  - **.summary**
- **.playerRankings**
  - **.runData**
- **.corrTbl**
  - **.R**
  - **.p**
  - **.RLCL**
  - **.RUCL**
  - **.RSq**
  - **.RSqLCL**
  - **.RSqUCL**
  - **.RSqRange**
- **.winningRoutesTbl**
  - .nPlayersAll
  - .nPlayers2
  - .nPlayers4

Results struct from CompareResults.m: Level 1-3 Fields

- .fileID
- .file
- .settings
- **.playerStatsTbl**
  - **.runData**
    - .nPlayersAll
    - .nPlayers2
    - .nPlayers3
    - .nPlayers4
  - **.summary**
    - .nPlayersAll
    - .nPlayers2
    - .nPlayers4
- **.playerConfIntTbl**
  - **.runData**
    - .nPlayersAll
    - .nPlayers2
    - .nPlayers3
    - .nPlayers4
  - **.summary**
    - .nPlayersAll
    - .nPlayers2
    - .nPlayers4
- **.winRates**
  - **.runData**
    - .nPlayersAll
    - .nPlayers2
    - .nPlayers3
    - .nPlayers4
  - **.deviations**
    - .nPlayers2
    - .nPlayers4
- **.winRatesStatResultsTbl**
  - .nPlayersAll
  - .nPlayers2
  - .nPlayers4
- **.avgDiffPerRank**
  - **.runData**
    - .nPlayersAll
    - .nPlayers2
    - .nPlayers4
  - **.summary**
    - .nPlayers2
    - .nPlayers4
- **.playerRankings**
  - **.runData**
    - .nPlayersAll
    - .nPlayers2
    - .nPlayers3
    - .nPlayers4
- **.corrTbl**
  - **.R**
  - **.p**
  - **.RLCL**
  - **.RUCL**
  - **.RSq**
  - **.RSqLCL**
  - **.RSqUCL**
  - **.RSqRange**
- **.winningRoutesTbl**
  - .nPlayersAll
  - .nPlayers2
  - .nPlayers4

The correlation table has two additional levels structured as below. The other correlation table fields besides .R (i.e., .p, .RLCL, .RUCL, .RSq, .RSqLCL, .RSqUCL, and .RSqRange) are structured similarly.

Correlation Table Fields (including Levels 4 and 5)
.corrTbl
.R
.runData
.nPlayersAll
.nPlayers2
.nPlayers3
.nPlayers4
.deviations
.nPlayers2
.nPlayers3
.nPlayers4

A Few Compare Results Insights

As can be seen from the similar structure of the compareResults struct to the results output by the simulation games, most of the metrics are the same as those of the individual game results files, except that they are aggregated with results of multiple simulations to allow for results comparison at different parameter settings. Several metrics have a .runData field which can allow the user to see the summary results of all games regardless of the number of players (i.e., nPlayersAll) as well as directly compare games of the same number of players (e.g., nPlayers2). The idea behind this flexibility is that the key measures of this study could not be appropriately compared between two, three, and four player games (e.g., deviations from the win rate would be different from 2 vs. 3 players because the ideal win rates would be 50% versus 33%, respectively.

Several metrics also have a summary or deviations field to allow results from simulations to be more concise and easier to interpret and relate as a whole (rather than broken down by player). For example, the .winRates field in the compareResults struct has a .deviations field that allows for easier comparison on the "fairness" or balance of games. It measures the deviation of player's win rates from an equal win rate among players, allowing for side-by-side comparison between the games. The results would suggest that whichever game has an average deviation closest to 0, could be the most balanced game. While using the CompareResults function was not a large focus in this paper, it nonetheless allows researchers an additional tool for interpreting and comparing the results of different simulations should they choose to run or expand upon this study.

Random Number Generation

Random number generation was used in the current study to model the random processes that occur during the Ticket to Game gameplay. Pseudorandom number generation was necessary to model shuffling of the train card deck and destination cards deck.

The MATLAB app UI allows the user the option to select between a “Consistent,” or default, and a “Shuffle” or random seed. The default option seeds the random number stream to 0. This option was given to allow for the reproducibility of results should someone want to replicate the simulation at the exact same settings and see “consistent” or the same results in the future (e.g., for the purpose of debugging or verifying results of the current study). The random seed allows the flexibility of varying the seed as needed.

To process hundreds of iterations more efficiently, the functionality of multiple parallel workers (or processing threads) executing separate games was implemented. This was controlled from the main program that controlled the running of the simulation in a parallel for loop. With multiple parallel threads, execution of tasks is not necessarily known ahead of time in the same order. To allow us to reproduce our results with the “Consistent” seed, it was necessary to instantiate parallel pseudorandom number streams and pass them separately to workers executing separate games.

Thus, for every game, the random stream was passed as an argument to each call to the function that simulated the game. This ensured that train decks from different games saw a consistent stream of random numbers if execution switched between threads. The random stream had to be passed to the Player takeTurn function since several of the player actions required the train deck, which required periodic shuffling when it became empty. This was necessary when players chose to claim a route, pick train cards, sacrifice a train, block a route, and steal.

There are four available pseudorandom number algorithms that support multiple streams and substream processing in MATLAB, and the one utilized in the current study is “threefry.” The period of “threefry” is 2^{514} or 2^{256} streams of length 2^{258}, which is the longest period available for the four algorithms (MathWorks, RNG 2022).

Model Verification

A set of MATLAB unit tests were created for the non-trivial functions of the base game components including the Board, TrainsDeck, Rules, and Player classes. These can be found in the “tests” directory and can be run with the “runtests” command on the MATLAB command line.

Additionally, a logging utility was created to log all player actions to the MATLAB command window, which allows for manual verification. The logger was created using an existing utility found on the MATLAB file exchange (Winslow 2022).

Performance and Optimization

Efforts were made to optimize the speed of the simulation so that it would be possible to run many iterations in a reasonable amount of time.

Wherever possible, functions were "vectorized" instead of using loops in order to take advantage of the parallelism provided by the built-in MATLAB libraries. For example, the getNumOfTrains method in the Board class is meant to return how many of a given number of train pieces are on the board. A loop-based implementation may look like this:

```
numTrains = 0;
numRoutes = length(board.initialRoutes);
for iRoute = 1:numRoutes
    if board.routeGraph.Edges.('Owner')(iRoute) == color
        numTrains = numTrains + board.routeGraph.Edges.('Length')(iRoute);
    end
end
```

The following is the vectorized implementation of the same logic:

```
numTrains = sum(board.routeGraph.Edges.Length(board.routeGraph.Edges.Owner==color));
```

According to the MATLAB profiler, the vectorized implementation is about 45 times faster.

The speed of the overall simulation depends on the player agents being used, as calculating the route utilities for the utility-based strategic players is by far the most computationally expensive part of the simulation. The MATLAB Parallel Computing Toolbox was used to parallelize the top-level iteration loop in RunSimulation.m so that multiple games could be simulated at the same time. With this parallelization, the simulation can run hundreds of game iterations in minutes.

Experimental Results and Validation

Results in tables are reported as "M [LL UL]" where M is the mean value and LL and UL are the lower limit and upper limit of the 95% confidence interval respectively.

Long Route Player vs. Destination Ticket Player

Experiments were carried out to evaluate games between players employing two broadly different strategies. The Long Route player is a utility-based strategic player with a Long Route utility weight of 1.0, meaning they are solely focused on claiming long routes and are not concerned with completing destination tickets. The Destination Ticket player is a utility-based strategic player with a Destination Ticket utility weight of 0.9 and a Long Route utility weight of 0.1 (so that they will still claim routes after completing all destination tickets).

Experiment 1: 2-player Baseline
Player 1: LongRoutePlayer
Player 2: DestinationTicketPlayer
Rules: DefaultRules
Board: Base game board
Iterations: 200

	Player 1	***Player 2***
Score	90.00 [88.95 91.05]	59.82 [56.87 62.77]
Trains Played	44.52 [44.43 44.61]	28.94 [28.54 29.33]
Train Cards Left in Hand	3.49 [2.98 4.00]	4.57 [4.10 5.03]
Destination Ticket Cards Completed	0 [0 0]	2.46 [2.36 2.55]
Turns Played	31.15 [30.90 31.40]	30.15 [29.90 30.40]
Number of Routes Claimed	8.72 [8.61 8.83]	11.72 [11.53 11.90]
Average Claimed Route Length	5.14 [5.08 5.20]	2.49 [2.45 2.53]
Longest Route Length	17.44 [16.80 18.08]	18.71 [18.04 19.37]
Number of Turns in Lead	21.60 [21.00 22.20]	9.89 [9.36 10.42]
Number of Turns in Lead Including Longest Route Bonus	17.83 [17.21 18.45]	13.55 [12.96 14.13]
Win Rate	0.87	0.13

	1st vs. 2nd
Average Score Difference	33.04

The LongRoutePlayer is clearly more successful than the DestinationTicketPlayer and the player statistics provide us insight as to why. The DestinationTicketPlayer claims on average 3 more routes than the LongRoutePlayer, but the average length of the routes claimed by the DestinationTicketPlayer is much smaller than the average length of the routes claimed by the LongRoutePlayer. The DestinationTicketPlayer claims small routes with the belief that their utility in completing destination ticket cards outweighs the utility of the points provided by claiming longer routes. The player statistics show, however, that the perceived utility of completing destination ticket cards does not translate to the player's final score. We can see this more clearly by examining some of the key correlation statistics.

	Correlation Coefficient	*p value*
Trains Played vs Score	0.77 [0.72 0.80]	2.45 E-78
Average Claimed Route Length vs Score	0.72 [0.67 0.76]	5.46E-65
Destination Tickets Completed vs Score	-0.51 [-0.58 -0.43]	1.29E-27
Number of Turns in Lead vs Score	0.59 [0.52 0.65]	1.31E-38
Number of Turns in Lead Including Longest Route Bonus vs Score	0.35 [0.26 0.43]	7.39E-13

The number of trains played and the average claimed route length are strongly correlated to the player's final score. The number of destination tickets completed is actually negatively correlated with player score, likely because the routes that are prioritized by the DestinationTicketPlayer have low point values and the points provided by completing destination ticket cards are not enough to make up for the potential points that are lost when trains are not used to claim larger high-scoring routes.

The simulation statistics show that the LongRoutePlayer's turns are more efficient at earning points than those of the DestinationTicketPlayer. With an average claimed route length of 5.14 trains, the LongRoutePlayer is theoretically able to play 43 trains and end the game in as little as 28 turns:
(43 cards – 4 starting cards)/(2 cards per turn)+(43 trains)/(5.14 trains per route) = 27.86 turns
This rough calculation assumes the cards the player needs to draw are always available though. The average number of turns taken by the LongRoutePlayer in this experiment is 31.15.

In contrast, the DestinationTicketPlayer, with an average claimed route length of 2.49, would take a theoretical minimum of 37 turns to play 43 trains and end the game. This experiment seems to show that a LongRoutePlayer is able to thwart the strategy of the DestinationTicketPlayer simply by ending the game before the DestinationTicketPlayer is able to play a reasonable number of trains to fully execute their strategy. We test this idea by simulating games between two DestinationTicketPlayers.

Experiment 2: 2 DestinationTicketPlayers
Player 1: DestinationTicketPlayer
Player 2: DestinationTicketPlayer
Rules: DefaultRules
Board: Base game board
Iterations: 200

	Player 1	*Player 2*
Score	108.22 [105.32 111.11]	105.23 [102.31 108.14]
Trains Played	42.94 [42.48 43.40]	42.19 [41.69 42.69]
Train Cards Left in Hand	6.82 [5.95 7.68]	7.35 [6.44 8.25]
Destination Ticket Cards Completed	3.59 [3.48 3.69]	3.46 [3.35 3.56]
Turns Played	40.44 [40.12 40.75]	39.88 [39.57 40.19]
Number of Routes Claimed	15.94 [15.71 16.173]	15.62 [15.38 15.85]
Average Claimed Route Length	2.71 [2.68 2.74]	2.72 [2.68 2.76]
Longest Route Length	26.93 [26.00 27.8]	26.36 [25.51 27.21]
Number of Turns in Lead	23.08 [21.81 24.34]	21.12 [19.81 22.42]
Number of Turns in Lead Including Longest Route Bonus	21.65 [20.30 23.00]	20.25 [18.86 21.64]
Win Rate	0.515	0.485

	1st vs. 2nd
Average Score Difference	28.15

Without a LongRoutePlayer, games take about 40 turns to end instead of about 31. With a longer game, DestinationTicketPlayers are able to claim about 4 additional routes and complete about 1 additional destination ticket. This is enough to increase the average DestinationTicketPlayer score from around 60 to over 100. The correlation statistics show that for this simulation, destination ticket completion is positively correlated with final score.

	Correlation Coefficient	*p value*
Trains Played vs Score	0.58 [0.51 0.64]	1.24E-37
Average Claimed Route Length vs Score	0.20 [0.10 0.29]	5.61E-5
Destination Tickets Completed vs Score	0.54 [0.47 0.61]	4.74E-32
Number of Turns in Lead vs Score	0.37 [0.28 0.45]	3.62E-14
Number of Turns in Lead Including Longest Route Bonus vs Score	0.30 [0.21 0.39]	9.86E-10

The results of Experiment 1 may indicate a problem to be addressed by the game designers, as it shows that players may be able to essentially ignore the destination ticket cards and score much higher than players who don't. Research has indicated that closeness of games and skill being correlated with game outcome are important factors in a game's enjoyability. Completing destination ticket cards is subjectively more challenging than simply claiming long routes, so the lack of correlation between destination ticket card completion and final score in games where the LongRoutePlayer is present may indicate an opportunity to improve the enjoyability of the game. The following experiments show how different rules variations may impact these goals.

Experiment 3: Destination Ticket Bonus
Player 1: LongRoutePlayer
Player 2: DestinationTicketPlayer
Rules: DestinationTicketBonusRules
Board: Base game board
Iterations: 200

This rules variation adds a bonus of 7 points for each destination ticket card completed.

	Player 1	***Player 2***
Score	90.00 [88.95 91.05]	77.00 [73.65 80.36]
Trains Played	44.52 [44.43 44.61]	28.94 [28.54 29.33]
Train Cards Left in Hand	3.49 [2.98 4.00]	4.57 [4.10 5.03]
Destination Ticket Cards Completed	0 [0 0]	2.46 [2.36 2.55]
Turns Played	31.15 [30.90 31.40]	30.15 [29.90 30.40]
Number of Routes Claimed	8.72 [8.61 8.83]	11.72 [11.53 11.90]
Average Claimed Route Length	5.14 [5.08 5.20]	2.49 [2.45 2.53]
Longest Route Length	17.44 [16.80 18.08]	18.71 [18.04 19.37]
Number of Turns in Lead	21.60 [21.00 22.20]	9.89 [9.36 10.42]
Number of Turns in Lead Including Longest Route Bonus	17.83 [17.21 18.45]	13.55 [12.96 14.13]
Win Rate	0.715	0.285

	1st vs. 2nd
Average Score Difference	24.66

Since completed destination ticket cards are scored at the end, this variation only impacts the final score player statistics and win rate compared to Experiment 1. This rules variation allows the DestinationTicketPlayer to score more points and improves their win rate against the LongRoutePlayer, but the LongRoutePlayer still spends much of the game in the lead and the average final score differential is still large. This variation does, however, make the correlation between destination tickets completed and final score less negative. It appears the DestinationTicketPlayer is still not able reliably accumulate enough points to win before the LongRoutePlayer ends the game.

	Correlation Coefficient	***p value***
Trains Played vs Score	0.45 [0.37 0.53]	1.09E-21
Average Claimed Route Length vs Score	0.39 [0.30 0.47]	5.95E-16
Destination Tickets Completed vs Score	-0.10 [-0.20 0]	0.0482
Number of Turns in Lead vs Score	0.80 [0.77 0.84]	3.85E-92
Number of Turns in Lead Including Longest Route Bonus vs Score	0.46 [0.38 0.53]	3.51E-22

Experiment 4: Less Route Points
Player 1: LongRoutePlayer
Player 2: DestinationTicketPlayer
Rules: LessRoutePointsRules
Board: Base game board
Iterations: 200

This rule variation decreases the number of points players earn just for claiming routes

	Player 1	***Player 2***
Score	56.39 [55.43 57.35]	52.43 [49.64 55.22]
Trains Played	44.52 [44.43 44.61]	28.94 [28.54 29.33]
Train Cards Left in Hand	3.49 [2.98 4.00]	4.57 [4.10 5.03]
Destination Ticket Cards Completed	0 [0 0]	2.46 [2.36 2.55]
Turns Played	31.15 [30.90 31.40]	30.15 [29.90 30.40]
Number of Routes Claimed	8.72 [8.61 8.83]	11.72 [11.53 11.90]
Average Claimed Route Length	5.14 [5.08 5.20]	2.49 [2.45 2.53]
Longest Route Length	17.44 [16.80 18.08]	18.71 [18.04 19.37]
Number of Turns in Lead	20.06 [19.44 10.68]	11.85 [11.28 12.41]
Number of Turns in Lead Including Longest Route Bonus	14.95 [14.33 15.56]	16.62 [16.03 17.21]
Win Rate	0.555	0.445

	1st vs. 2nd
Average Score Difference	18.99

The results of this simulation seem to show that this rules variation can provide the game with a better “balance” between the different strategies. The win rates are closer, the number of turns each player spends in the lead are closer to equal, and the average final score differential is smaller. If the game designers desired for both the LongRoutePlayer’s and DestinationTicketPlayer’s strategies to be about equally viable, it appears that this variation could achieve that.

	Correlation Coefficient	*p value*
Trains Played vs Score	0.25 [0.16 0.34]	4.02E-7
Average Claimed Route Length vs Score	0.18 [0.08 0.27]	3.80E-4
Destination Tickets Completed vs Score	0.08 [-0.02 0.18]	0.12
Number of Turns in Lead vs Score	0.12 [0.02 0.22]	0.016
Number of Turns in Lead Including Longest Route Bonus vs Score	0.06 [-0.04 0.15]	0.27

The correlation statistics show a decreased correlation between the number of trains played and final score as well as between the average claimed route length and final score as compared to Experiment 1. This makes sense as this rules variation gives much less reward for claiming just any long route. There is still no strong correlation between destination ticket completion and final score however, since the LongRoutePlayer is still able to score well without completing any destination ticket cards. Interestingly, this experiment results in very little correlation between the final score and the number of turns spent in the lead. This makes sense, as many of the DestinationTicketPlayer’s points are awarded at the end. This element of not knowing much of the final score before the very end could increase the game’s enjoyability, as research has indicated games are less fun when the winner is known long before the game ends.

Experiment 5: More Actions Per Turn
Player 1: LongRoutePlayer
Player 2: DestinationTicketPlayer
Rules: MoreActionsRules
Board: Base game board
Iterations: 200

This experiment uses rules which allow players to take any combination of three actions which include drawing cards or claiming routes (claim 3 routes, claim 1 route and draw 2 cards, claim 2 routes and draw 1 card, or draw 3 cards) on each turn.

	Player 1	***Player 2***
Score	85.16 [84.20 86.12]	79.36 [76.46 82.26]
Trains Played	44.73 [44.67 44.79]	35.12 [34.66 35.57]
Train Cards Left in Hand	2.86 [2.49 3.22]	3.35 [2.92 3.78]
Destination Ticket Cards Completed	0 [0 0]	2.96 [2.87 3.04]
Turns Played	17.83 [17.71 17.94]	16.83 [16.71 16.94]
Number of Routes Claimed	9.89 [9.76 10.02]	14.42 [14.20 14.64]
Average Claimed Route Length	4.56 [4.50 4.63]	2.45 [2.42 2.48]
Longest Route Length	16.62 [15.97 17.27]	21.58 [20.86 22.29]
Number of Turns in Lead	10.63 [10.24 11.00]	6.82 [6.48 7.15]
Number of Turns in Lead Including Longest Route Bonus	8.33 [7.98 8.68]	8.93 [8.62 9.24]
Win Rate	0.57	0.43

	1st vs. 2nd
Average Score Difference	20.41

While not perfectly balanced, this variation does allow the DestinationTicketPlayer to win more games and complete an average of 0.5 more destination tickets per game. Games are about 13-14 turns shorter and both players spend a similar amount of time in the lead when the longest route bonus is taken into account.

	Correlation Coefficient	***p value***
Trains Played vs Score	0.44 [0.35 0.51]	4.74E-20
Average Claimed Route Length vs Score	0.25 [0.15 0.34]	6.14E-7
Destination Tickets Completed vs Score	-0.06 [-0.15 0.04]	0.26
Number of Turns in Lead vs Score	0.55 [0.48 0.61]	1.04E-32
Number of Turns in Lead Including Longest Route Bonus vs Score	-0.09 [-0.20 -0.01]	0.07

The difference between the correlation of Number of Turns in Lead vs Score and Number of Turns in Lead Including Longest Route Bonus vs Score makes sense when looking at the player statistics, as both players generally spend the same amount of turns in the lead only when the Longest Route Bonus is taken into account, but the players end up with significantly different scores.

Experiment 6: All Routes Same Length
Player 1: LongRoutePlayer
Player 2: DestinationTicketPlayer
Rules: DefaultRules
Board: Same Route Length board
Iterations: 200

This experiment uses the default rules, but also uses a board where all of the routes are three trains long. Points for completing destination ticket cards were not adjusted.

	Player 1	***Player 2***
Score	44.75 [43.84 45.65]	100.22 [97.10 103.34]
Trains Played	44.87 [44.75 44.98]	39.03 [38.32 39.74]
Train Cards Left in Hand	12.52 [11.49 13.55]	9.86 [8.68 11.03]
Destination Ticket Cards Completed	0 [0 0]	3.52 [3.41 3.64]
Turns Played	41.55 [41.07 42.03]	40.65 [40.16 41.14]
Number of Routes Claimed	14.96 [14.92 14.99]	13.01 [12.77 13.25]
Average Claimed Route Length	3 [3 3]	3 [3 3]
Longest Route Length	19.74 [19.26 20.22]	24.26 [23.51 25.00]
Number of Turns in Lead	30.09 [28.75 31.42]	23.77 [21.98 25.56]
Number of Turns in Lead Including Longest Route Bonus	19.25 [17.88 20.61]	27.25 [25.66 28.85]
Win Rate	0.03	0.97

	1st vs. 2nd
Average Score Difference	56.20

Unsurprisingly, the DestinationTicketPlayer is much more effective in this simulation. The LongRoutePlayer is able to claim more routes as they don't need to care about claiming specific routes, but they are not able to claim enough extra routes to make up for the points they lose from not completing destination tickets. The game takes about 10 more turns than the game using the default board, giving the DestinationTicketPlayer time to complete an average of 1 more destination ticket.

	Correlation Coefficient	*p value*
Trains Played vs Score	-0.28 [-0.37 -0.19]	7.51E-9
Average Claimed Route Length vs Score	N/A	N/A
Destination Tickets Completed vs Score	0.94 [0.93 0.95]	6.67E-194
Number of Turns in Lead vs Score	-0.04 [-0.14 0.06]	0.44
Number of Turns in Lead Including Longest Route Bonus vs Score	0.52 [0.45 0.59]	3.91E-29

The lack of correlation between the Number of Turns in Lead vs Score can be explained by the fact that with all of the routes being the same length, both players are able to claim routes at a similar rate, so they have a similar score until the very end when points for completing destination tickets and the longest route bonus are applied. In this experiment, final score was very strongly correlated with destination ticket completion but was negatively correlated with the number of trains played. This variation could be used if the game designers wanted to ensure that completing destination tickets was the main focus of the game.

Experiment 7: 4 Player Simulation
Player 1: LongRoutePlayer
Player 2: DestinationTicketPlayer
Player 3: DestinationTicketPlayer
Player 4: DestinationTicketPlayer
Rules: DefaultRules
Board: Base game board
Iterations: 200

	Player 1	*Player 2*	*Player 3*	*Player 4*
Score	84.82 [83.95 85.69]	63.63 [60.48 66.78]	65.16 [61.82 68.50]	63.81 [60.59 67.04]
Trains Played	44.56 [44.47 44.65]	30.73 [30.26 31.20]	31.58 [31.05 32.10]	31.23 [30.73 31.73]
Train Cards Left in Hand	4.12 [3.53 4.70]	5.43 [4.90 5.95]	5.41 [4.86 5.96]	5.83 [5.22 6.43]
Destination Ticket Cards Completed	0 [0 0]	2.56 [2.46 2.66]	2.47 [2.36 2.58]	2.53 [2.42 2.63]
Turns Played	31.88 [31.59 32.16]	30.88 [30.59 31.17]	30.88 [30.59 31.17]	30.88 [30.59 31.16]
Number of Routes Claimed	9.1 [9.00 9.20]	11.95 [11.76 12.13]	12.09 [11.91 12.27]	12.25 [12.06 12.43]
Average Claimed Route Length	4.93 [4.87 4.99]	2.59 [2.55 2.63]	2.62 [2.58 2.67]	2.56 [2.52 2.60]
Longest Route Length	16.41 [15.66 17.16]	19.92 [19.16 20.67]	20.06 [19.34 20.78]	19.98 [19.24 20.72]
Number of Turns in Lead	21.17 [20.55 21.78]	4.46 [3.98 4.94]	3.98 [3.54 4.41]	4.34 [3.87 4.81]
Number of Turns in Lead Including Longest Route Bonus	17.27 [16.62 17.91]	5.24 [4.73 5.74]	4.94 [4.38 5.49]	5.59 [5.06 6.12]
Win Rate	0.52	0.14	0.185	0.155

	1st vs. 2nd	***2nd vs 3rd***	***3rd vs 4th***
Average Score Difference	17.08	14.55	15.55

This experiment shows that in a 4-player game with one LongRoutePlayer, the LongRoutePlayer scores slightly lower than in the 2-player experiment, but still beats the DestinationTicketPlayers the majority of the time. The results are generally very similar to the results of Experiment 1 for each type of player. This shows that the addition of more DestinationTicketPlayers does not significantly impact the strategy of the LongRoutePlayer.

Other Player Agents

Experiment 1: Deviant Player

Player 1: LongRoutePlayer
Player 2: DestinationTicketPlayer
Player 3: DeviantPlayer
Rules: DefaultRules
Board: Base game board

Iterations: 200

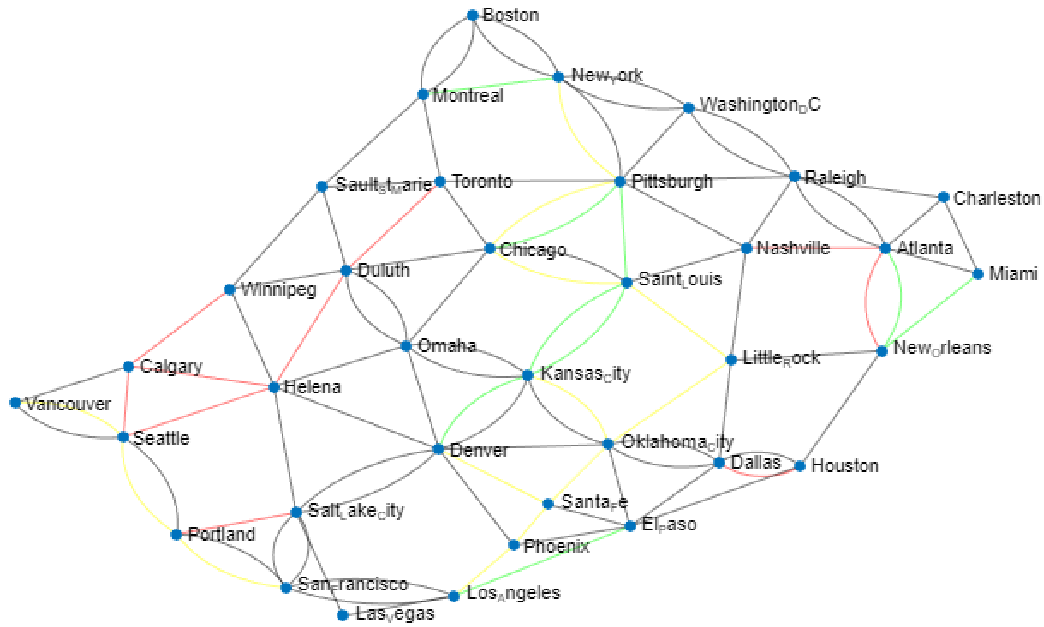
The DeviantPlayer is a utility-based strategic player with a deviant utility weight of 1.0. The DeviantPlayer attempts to sabotage the strategies of the other players by blocking them from claiming long routes or completing destination tickets.

	Player 1	***Player 2***	***Player 3***
Score	80.88 [79.82 81.93]	58.08 [55.01 61.14]	59.19 [57.70 60.69]
Trains Played	44.06 [43.94 44.17]	30.04 [29.64 30.44]	39.13 [38.74 39.53]
Train Cards Left in Hand	1.79 [1.50 2.07]	4.96 [4.52 5.40]	2.42 [2.14 2.69]
Destination Ticket Cards Completed	0 [0 0]	2.37 [2.26 2.47]	0.01 [0 0.024]
Turns Played	31.51 [31.34 31.68]	30.56 [30.39 30.73]	30.56 [30.39 30.73]
Number of Routes Claimed	9.92 [9.81 10.02]	12.35 [12.18 12.52]	10.05 [9.93 10.17]
Average Claimed Route Length	4.47 [4.42 4.51]	2.45 [2.41 2.49]	3.92 [3.86 3.98]
Longest Route Length	16.76 [16.01 17.51]	18.73 [18.05 19.40]	13.32 [12.77 13.87]
Number of Turns in Lead	18.98 [18.20 19.75]	6.05 [5.70 6.40]	11.27 [10.46 12.07]
Number of Turns in Lead Including Longest Route Bonus	16.85 [16.07 17.63]	8.81 [8.29 9.33]	9.06 [8.29 9.83]
Win Rate	0.75	0.195	0.055

	1st vs. 2nd	***2nd vs 3rd***
Average Score Difference	16.85	17.8

The DeviantPlayer does not appear to be very effective at preventing other players from carrying out their strategies. The average claimed route length for the LongRoutePlayer is about 0.5 less in this simulation than in the simulations between a LongRoutePlayer and DestinationTicketPlayers, indicating that the DeviantPlayer is occasionally able to block the LongRoutePlayer from claiming long routes, but this does not appear to significantly impact the LongRoutePlayer's final score or ability to win games. The number of destination tickets completed by the DestinationTicketPlayer in this simulation is not significantly different than in the simulations without a

DeviantPlayer, indicating that the DeviantPlayer is not successful at blocking the DestinationTicketPlayer from completing tickets.



Graph of the final board state for a simulated game in Experiment 1. Red is a LongRoutePlayer and yellow is a DestinationTicketPlayer, and green is a DeviantPlayer. One can see where the DeviantPlayer has claimed routes connected to routes claimed by other players, attempting to block destination ticket card completion.

	Correlation Coefficient	***p value***
Trains Played vs Score	0.61 [0.5602 0.6604]	3.94E-63
Average Claimed Route Length vs Score	0.53 [0.4701 0.5854]	8.67E-45
Destination Tickets Completed vs Score	-0.14 [-0.21 -0.06]	9.15E-4
Number of Turns in Lead vs Score	0.49 [0.42 0.55]	2.94E-37
Number of Turns in Lead Including Longest Route Bonus vs Score	0.48 [0.42 0.54]	1.95E-36

The correlation statistics are not very different from games between only LongRoutePlayers and DestinationTicketPlayers. Trains Played and Average Claimed Route Length are a bit less correlated with final score, and Destination Ticket completion is less negatively correlated, likely because the DeviantPlayer claims some long routes and completes no destination cards, but generally scores quite a bit lower than the LongRoutePlayer.

The DeviantPlayer's task is difficult, as it does not know which destination cards each player has in their hand, what color train cards each player has in their hand, or which strategy each player is using. The DeviantPlayer attempts to block long routes by observing the color of train cards the other players draw, but the other players may draw cards from the top of the train card deck which are hidden from the DeviantPlayer. If the DeviantPlayer does successfully block a long route by claiming it, there is usually still a long route of that same color for the LongRoutePlayer to claim, so the loss is not great. For blocking destination ticket completion, there are simply too many paths for other players to use to complete their tickets that it is nearly impossible for the DeviantPlayer to block all reasonable paths, especially when the DeviantPlayer cannot know which destination tickets the other players are trying to complete. This experiment indicates that the game rules or board structure would require modification for deviant play to be possible.

Experiment 2: Hybrid Player
Player 1: LongRoutePlayer
Player 2: HybridPlayer
Rules: DefaultRules
Board: Base game board
Iterations: 200

The HybridPlayer is a utility-based strategic player with a Long Route utility weight of 0.5 and a Destination Ticket utility weight of 0.5. This player is trying to complete destination tickets but is also considering the length of the routes they choose to claim.

	Player 1	***Player 2***
Score	86.18 [85.09 87.27]	76.31 [73.66 78.96]
Trains Played	44.44 [44.34 44.53]	32.22 [31.79 32.65]
Train Cards Left in Hand	2.94 [2.47 3.40]	2.94 [2.57 3.30]
Destination Ticket Cards Completed	0 [0 0]	1.99 [1.90 2.08]
Turns Played	31.37 [31.13 31.60]	30.37 [30.13 30.60]
Number of Routes Claimed	9.17 [9.05 9.28]	11.08 [10.90 11.25]
Average Claimed Route Length	4.89 [4.83 4.96]	2.94 [2.88 3.00]
Longest Route Length	16.49 [15.80 17.17]	20.96 [20.28 21.63]
Number of Turns in Lead	22.02 [21.43 22.61]	13.39 [12.81 13.96]
Number of Turns in Lead Including Longest Route Bonus	18.90 [18.28 19.51]	16.33 [15.73 16.93]
Win Rate	0.6350	0.3650

	1st vs. 2nd
Average Score Difference	20.15

The HybridPlayer fares much better against the LongRoutePlayer than the DestinationTicketPlayer and is still able to complete an average of almost 2 destination tickets per game. The LongRoutePlayer still wins a majority of the time, but the games are closer, and the correlation statistics show a weaker correlation between number of turns in the lead and final score.

	Correlation Coefficient	*p value*
Trains Played vs Score	0.46 [0.38 0.54]	1.23E-22
Average Claimed Route Length vs Score	0.39 [0.30 0.47]	5.84E-16
Destination Tickets Completed vs Score	-0.12 [-0.21 -0.02]	0.02
Number of Turns in Lead vs Score	0.32 [0.22 0.40]	1.06E-10
Number of Turns in Lead Including Longest Route Bonus vs Score	0.21 [0.12 0.30]	2.04E-5

This experiment indicates that a strategy focusing on both long routes and completing destination tickets may not be as effective as a strategy simply focusing on long routes. The HybridPlayer could be considered to have a much more complicated strategy than the LongRoutePlayer and the fact that this more complicated strategy is not rewarded with a better outcome in the game could impact the game's enjoyability.

Experiment 3: Varied Players

Player 1: LongRoutePlayer
Player 2: DestinationTicketPlayer
Player 3: DeviantPlayer
Player 4: HybridPlayer
Rules: DefaultRules
Board: Base game board
Iterations: 200

	Player 1	*Player 2*	*Player 3*	*Player 4*
Score	78.04 [77.02 79.05]	58.01 [55.03 60.99]	54.54 [53.01 56.07]	73.12 [70.17 76.06]
Trains Played	44.30 [44.19 44.40]	30.81 [30.42 31.20]	38.70 [38.30 39.09]	33.84 [33.43 34.25]
Train Cards Left in Hand	1.82 [1.56 2.07]	5.08 [4.61 5.54]	2.74 [2.38 3.09]	3.32 [2.91 3.73]
Destination Ticket Cards Completed	0.01 [0 0.01]	2.36 [2.25 2.46]	0 [0 0]	1.85 [1.75 1.95]
Turns Played	31.85 [31.68 32.02]	30.88 [30.71 31.05]	30.88 [30.71 31.05]	30.85 [30.68 31.02]
Number of Routes Claimed	10.04 [9.94 10.14]	12.41 [12.24 12.57]	10.52 [10.37 10.66]	11.72 [11.56 11.87]
Average Claimed Route Length	4.43 [4.39 4.48]	2.50 [2.46 2.53]	3.71 [3.65 3.77]	2.91 [2.86 2.96]
Longest Route Length	16.70 [15.95 17.44]	19.26 [18.55 19.97]	13.52 [13.00 14.03]	21.35 [20.61 22.08]
Number of Turns in Lead	18.77 [17.98 19.55]	4.46 [4.17 4.75]	8.14 [7.29 8.98]	5.77 [5.22 6.31]
Number of Turns in Lead Including Longest Route Bonus	16.34 [15.55 17.13]	6.39 [5.94 6.83]	6.05 [5.31 6.79]	6.87 [6.21 7.52]
Win Rate	0.4350	0.1650	0.02	0.38

	1^{st} vs. 2^{nd}	*2^{nd} vs 3^{rd}*	*3^{rd} vs 4^{th}*
Average Score Difference	14.64	12.82	15.32

In a game between 4 different strategic players, the LongRoutePlayer and HybridPlayer are the most successful in terms of wins. Interestingly, the LongRoutePlayer spends by far the most amount of time in the lead but only wins 43.5% of games. This indicates that points awarded at the end of the game for destination ticket completion have a significant impact on the game's outcome. Indeed, the correlation statistics now show a positive correlation between destination ticket completion and final score.

	Correlation Coefficient	*p value*
Trains Played vs Score	0.40 [0.34 0.46]	1.07E-32
Average Claimed Route Length vs Score	0.33 [0.27 0.39]	3.45E-22
Destination Tickets Completed vs Score	0.13 [0.07 0.20]	1.41E-4
Number of Turns in Lead vs Score	0.34 [0.28 0.40]	9.87E-24

Number of Turns in Lead Including Longest Route Bonus vs Score	0.38 [0.32 0.44]	8.30E-29

The average claimed route length for the LongRoutePlayer is about 0.5 shorter than in the games between a LongRoutePlayer and DestinationTicketPlayer. This matches the value seen in the Deviant Player experiment, so it appears that the DeviantPlayer, while not very successful at winning the game, can reduce the effectiveness of the LongRoutePlayer and potentially increase the enjoyment of the other strategic players by increasing their effectiveness.

Experiment 4: Treacherous Player
Player 1: Long Route Player
Player 2: DestinationTicketPlayer
Player 3: HybridPlayer
Player 4: TreacherousPlayer
Rule: TreacheryRules
Board: Base game board
Iterations: 200

	Player 1	*Player 2*	*Player 3*	*Player 4*
Score	81.44 [80.54 82.34]	63.30 [60.34 66.25]	78.18 [75.20 81.16]	-18.18 [-19.07 -17.28]
Trains Played	44.47 [44.37 44.57]	30.85 [30.44 31.26]	33.87 [33.49 34.24]	0.36 [0.11 0.61]
Train Cards Left in Hand	2.72 [2.35 3.09]	4.76 [4.33 5.19]	3.64 [3.21 4.06]	65.29 [64.86 65.72]
Destination Ticket Cards Completed	0 [0 0]	2.56 [2.46 2.65]	1.97 [1.88 2.06]	0 [0 0]
Turns Played	31.90 [31.70 32.09]	30.90 [30.70 31.09]	30.90 [30.70 31.09]	30.90 [30.70 31.09]
Number of Routes Claimed	9.64 [9.55 9.73]	12.29 [12.12 12.46]	11.28 [11.10 11.46]	0.06 [0.02 0.10]
Average Claimed Route Length	4.64 [4.59 4.68]	2.53 [2.49 2.57]	3.04 [2.98 3.09]	0.27 [0.10 0.44]
Longest Route Length	16.61 [15.93 17.29]	19.39 [18.65 20.12]	21.73 [21.04 22.41]	0.27 [0.10 0.44]
Number of Turns in Lead	20.56 [19.94 21.18]	5.46 [5.07 5.85]	8.10 [7.44 8.76]	0.77 [0.61 0.93]
Number of Turns in Lead Including Longest Route Bonus	16.86 [16.21 17.50]	7.69 [7.16 8.21]	8.92 [8.20 9.64]	0.77 [0.61 0.93]
Win Rate	0.395	0.18	0.425	0

Interestingly, while the Treacherous Player fills a similar role to the Deviant Player, substituting the latter with the former producing an inverse effect on the effectiveness of the Long Route Player and Hybrid Player strategies. As compared to the game with the Deviant Player, when the Treacherous Player is leveraging its unique ruleset, the Long Route Player's strategy starts to fall apart and the Hybrid player is able to adapt to the new conditions of the opposition. The Destination Ticket Player appears to have approximately the same experience with the Treacherous Player as with the Deviant Player.

Early in the design phase, it was hypothesized that the Treacherous Player would make the Long Route Player less efficient by way of stealing routes that make up a long route from this player. It appears that this is indeed the effect of the Treacherous Player, and it is suspected that since points from completing destination tickets are instant while points for longest route are only at the end of the game, the Long Route Player is more susceptible to having their points taken away in the case that their routes are stolen by the Treacherous Player.

	1st vs. 2nd	*2nd vs 3rd*	*3rd vs 4th*
Average Score Difference	73.06	22.08	14.11

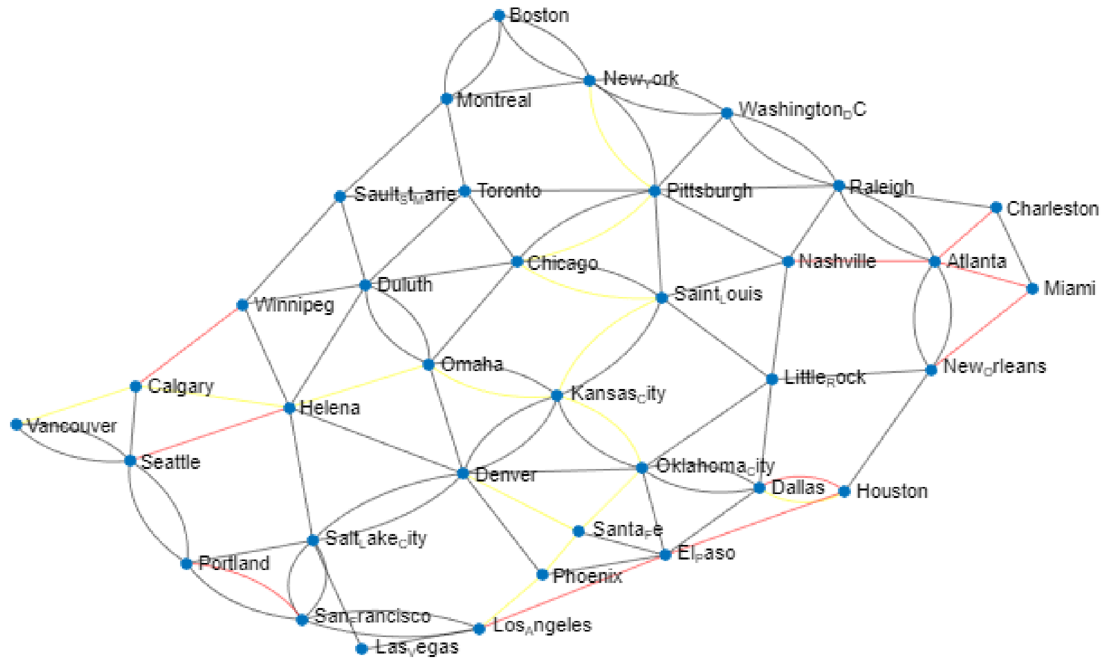
In addition to having an effect on the overall win rate of the players, the Treacherous Player has a significant effect on the differences between scores of players. While the difference in score between 3rd and 4th are fairly similar to the previous experiment, most of the average score differences have increased. This makes the games less close and increases the chance of a blow-out where one player is significantly ahead well before the end of the game.

	Correlation Coefficient	*p value*
Trains Played vs Score	0.93 [0.92 0.94]	0
Average Claimed Route Length vs Score	0.83 [0.80 0.85]	1.15E-200
Destination Tickets Completed vs Score	0.45 [0.40 0.51]	1.47E-41
Number of Turns in Lead vs Score	0.61 [0.57 0.66]	2.92E-84
Number of Turns in Lead Including Longest Route Bonus vs Score	0.68 [0.64 0.71]	8.38E-108

The correlation between the number of trains played and the final score is very high in this analysis; however, a large factor that is contributing to that is that the Treacherous Player plays hardly any trains and almost always ends with a negative score. It appears that the Treacherous Player succeeds in their objective of preventing other players from gaining points easily. However, since the gaps between ending scores increase, the enjoyment of the other players likely suffers from the introduction of the Treacherous Player into the game. This is quite similar to what one would expect from a real-world example of playing with a player who is attempting to foil everyone's objectives.

Player Agent Validation

The strategic player agents were qualitatively validated by manually examining the player activity log and final board state for simulated games. Members of the team are familiar with Ticket to Ride, having personally played many games themselves and so were able to provide "Face validation." In examining graphs of the final board state, it can be seen that the LongRoutePlayer claims long routes regardless of their location on the board, and the DestinationTicketPlayer claims routes to form connected paths. The paths the DestinationTicketPlayer forms to complete destination ticket cards appear reasonable, not containing loops or large deviations from the shortest routes.



Graph of the final board state for a simulated game. The colored edges indicate which player claimed that route. Red is a LongRoutePlayer and yellow is a DestinationTicketPlayer trying to complete the following destinations: Los Angeles to New York, Calgary to Phoenix, Chicago to Santa Fe.

In examining the player activity logs, it can be seen that the players draw cards which would complete their highest priority routes and claim those routes when they are able. The simulation statistics for each player were also qualitatively validated. The players in general do not contain an unreasonable number of cards in their hand at the end of the game and the DestinationTicketPlayer is able to complete a reasonable number of destination tickets.

Conclusion

Discussion

This project demonstrates how computer simulation can be used for thorough playtesting of a multiplayer turn-based strategy board game. The simulation provides insight into how the game is balanced, the effectiveness of different strategies, and metrics which may provide insight into the enjoyability of the game for different types of players. This simulation could not entirely replace playtesting with real-life players, but it would allow game designers to more quickly assess how different game variations may impact the game's balance and player experience.

Limitations of the Simulation

The simulation is of course limited in its ability to simulate a real-life game. Even with a game like TTR in which players can take only a limited number of actions, it is difficult to implement player agents that adequately model human players. Human players may not always follow a deterministic strategy, or their actions may be influenced by factors that are difficult to model, such as which areas of the board they are more likely to focus on.

Another limitation is that some aspects of a game's enjoyability are subjective or difficult to model in a computer simulation. For example, social interaction between players can be a factor in the players' enjoyment of the game, but it is difficult to measure if a game is conducive to social interaction using only computer simulation. This is another reason why a computer simulation could not fully replace real-life playtesting.

Future Work

Future work could include more rigorous validation of the player agent strategies using data from real-life games. Additional player agents could be implemented based on the observed strategies of real-life players. More sophisticated algorithms such as machine learning could be used to model player strategies more closely. Finally, a randomness element could be added to the choices made by the player agents to better model non-deterministic strategies of real-life players. All of this would serve to improve the validity of the player agents and the simulation as a whole.

Another opportunity for future work would be to expand on the metrics provided by the simulation to more thoroughly evaluate the experience of playing the game. This could include a way to measure the level of interaction between players' actions, which research indicates can be an important factor in the "fun-ness" of a game (Xu 2011)

Finally, an automated algorithm could be implemented to find the optimal values for game parameters (such as starting trains, route point values, etc.) which achieve desired metrics goals, effectively automating a portion of the game design process.

A summary video of the project can be viewed at this link:
https://mediaspace.gatech.edu/media/%5BTeam+37%5D+Ticket+to+Ride+Simulation/1_6l7qwhgl

References

Abuhamdeh, S., & Csikszentmihalyi, M. (2011). The Importance of Challenge for the Enjoyment of Intrinsically Motivated, Goal-Directed Activities. *Personality and Social Psychology Bulletin*, 317-330.

Barbara, J. (2014). Measuring User Experience in Board Games. *International Journal of Gaming and Computer-Mediated Simulations*, 64-79.

Barbara, J. (2017). Measuring User Experience in Multiplayer Board Games. *Games and Culture*, 623-649.

Chanel, G., Rebetez, C., Betrancourt, M., Run, T. (2011). Emotion Assessment From Physiological Signals for Adaptation of Game Difficulty. *IEEE Transactions on Systems, Man, and Cybernetics (2011).*

Guhe, M., & Lascarides, A. (2014). Game strategies for The Settlers of Catan. *2014 IEEE Conference on Computational Intelligence and Games.* Dortmund: IEEE.

Holmberg, M., & Modee, A. (2021). Random rewards in video games and their impact on player engagement. *Digitala Vetenskapliga Arkivet*.

Holmgard, C., Liapis, A., Togelius, J., & Yan-nakakis, G. N. (2014). Evolving personas for player decision modeling. *In Computational Intelligence and Games (CIG), 2014 IEEE Conference On*.

IJsselsteijn, W. A., de Kort, Y. A. W., & Poels, K. (2013). The Game Experience Questionnaire. Technische Universiteit Eindhoven.

Jaffe, A., Miller, A., Andersen, E., Liu, Y.-E., Karlin, A., & Popovi, Z. (2012). Evaluating Competitive Game Balance with Restricted Play. *AIIDE'12* (pp. 26-31). Stanford: AAAI Press.

Kirby, C. (n.d). 8 Types of Gamers You'll Meet Playing Board Games. Retrieved from https://www.underdoggames.com/a/blog/types-of-gamers

Luke Winslow (2022). log4m - A powerful and simple logger for matlab (https://www.mathworks.com/matlabcentral/fileexchange/37701-log4m-a-powerful-and-simple-logger-for-matlab), MATLAB Central File Exchange. Retrieved April 4, 2022.

MathWorks. (2022). rng. Retrieved from MathWorks: https://www.mathworks.com/help/matlab/ref/rng.html

Nutz, M., & Leifheit, L. (2021). Computer Science Unplugged: Developing and Evaluating a "Traveling Salesperson Problem" Board Game. *Proceedings of the 15th European Conference on Game-Based Learning*, (pp. 561-568).

Prensky, M. (2001). Fun, Play and Games: What Makes Games Engaging. In M. Prensky, *Digital-Based Learning* (pp. 1-31). New York: McGraw-Hill.

Roohi, S., Relas, A., Takatalo, J., Heiskanen, H., Hamalainen, P. (2020). Predicting Game Difficulty and Church Without Players. *Chi Play '20 Virtual Event, Paper Session 12: Investigating Player Behavior*.

Sargeantson, E. (n.d.). 19 Ticket to Ride Strategy Tips – How to Win! Retrieved from https://mykindofmeeple.com/ticket-to-ride-strategy-tips-how-to-win/

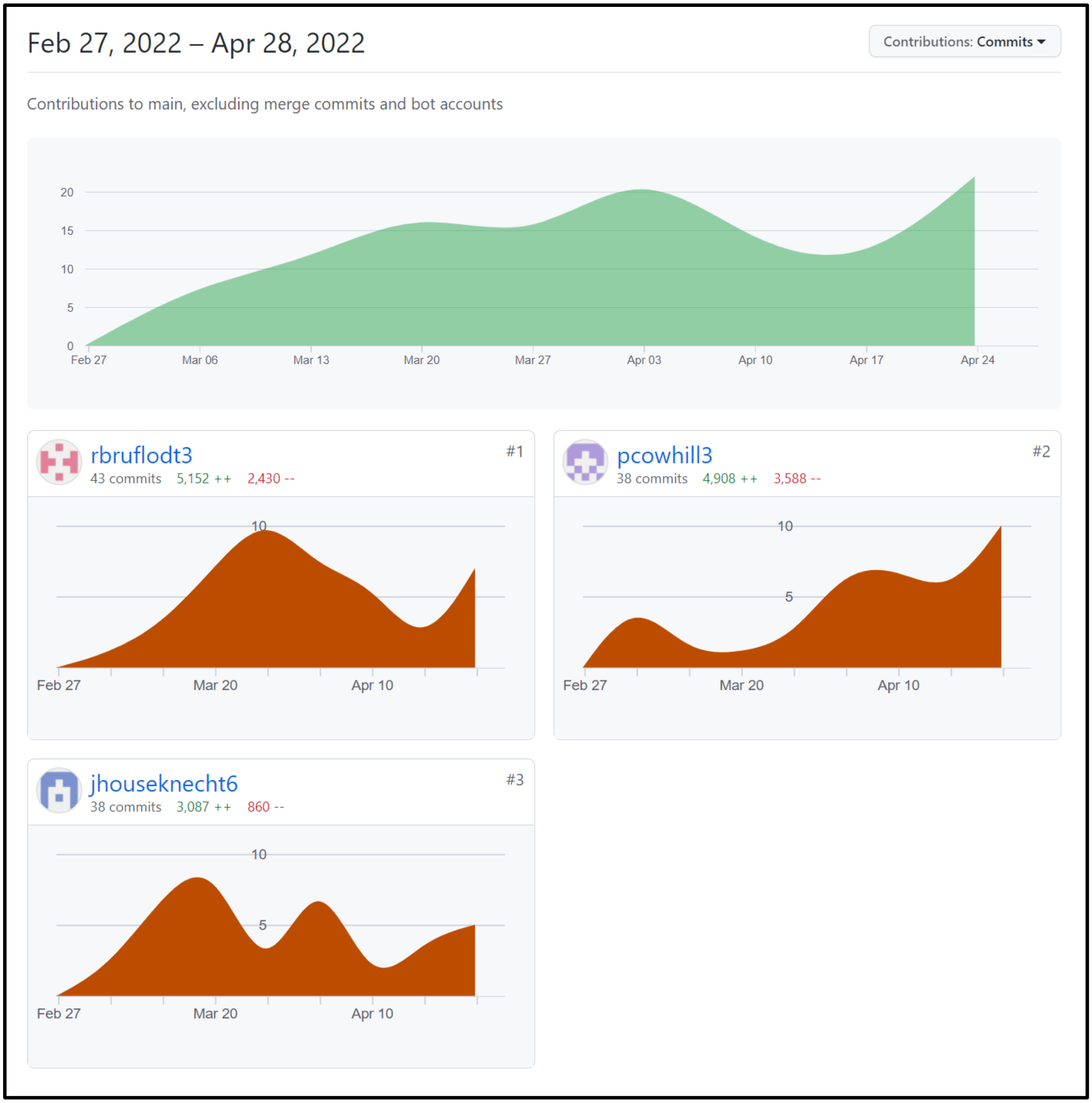
Silva, F. de M., Lee, S., Togelius, J., & Nealen, A. (2017). AI-based Playtesting of Contemporary Board Games. *Foundations of Digital Games (FDG)*.

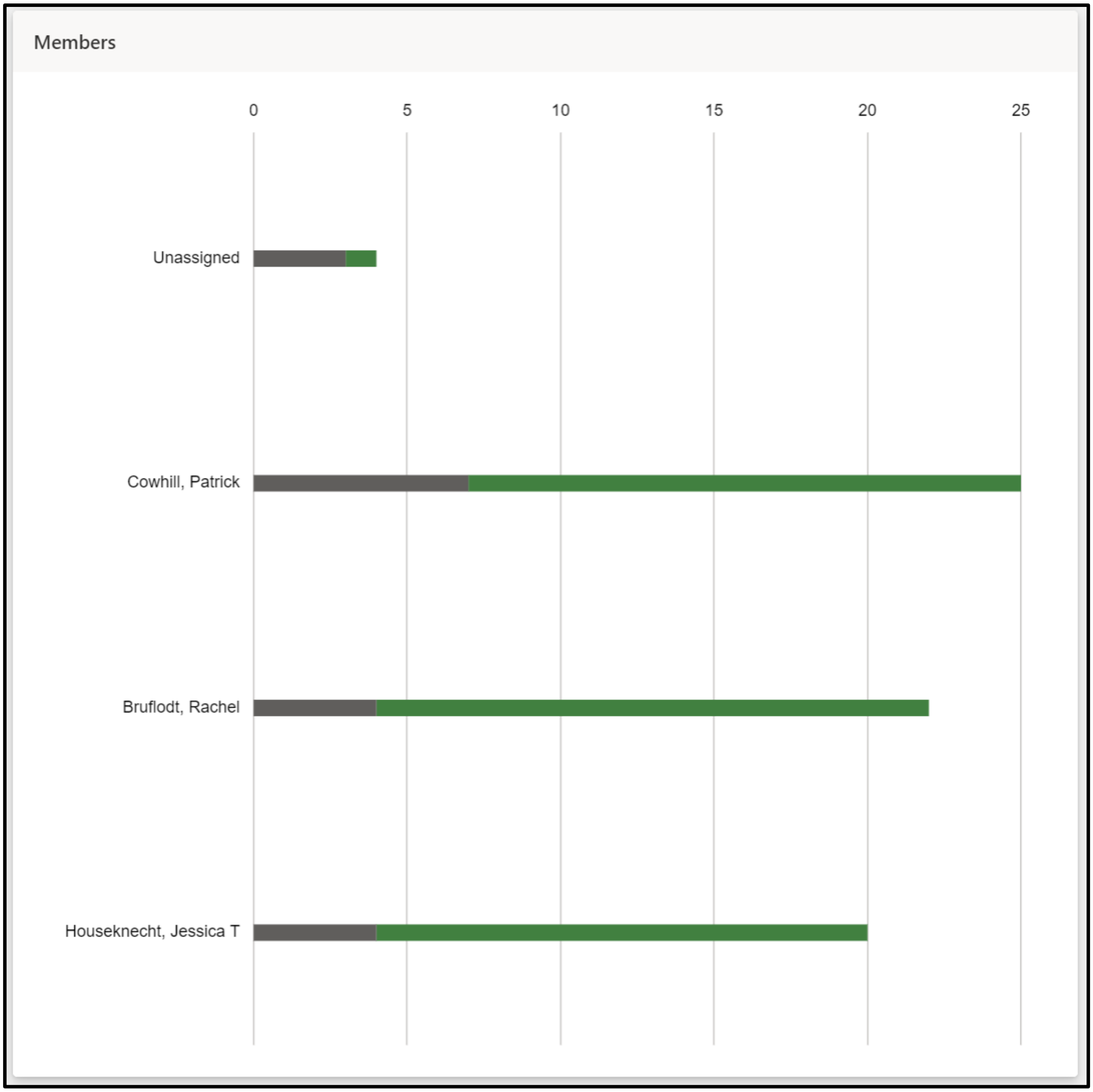
Xu, Y., Barba, E., Radu, I., Gandy, M., MacIntyre, B. (2011). Chares Are Fun: Understanding Social Play in Board Games for Digital Tabletop Game Design. *Georgia Institute of Technology.*

Appendix - Division of Labor

The team for this study met biweekly on Mondays and Thursdays to discuss updates from what has been worked on since the last meeting, concerns or blockers for upcoming assignments and milestones, and future goals to work towards until the next meeting. At these meetings, large milestones were broken up into lists of individual tasks that were then assigned to team members to complete by some deadline (which was usually the next meeting). Managing this list of tasks ensures that each team member was actively contributing every week to the development of this study and all portions of the study were being completed with minimal overlap between individuals. The tasks were either assigned to each team member based on their preferences, availability, and experience with a certain goal or were otherwise divided equally between the three members.

The tasks were managed in a scrum-style framework, and a collection of the current and completed tasks was automatically documented in MS Teams. MS Teams also has some analysis tools for its tasking feature that display various metrics of task allocation and completion, and this shows that there is an even division of labor between the team members. Additionally, code commits to the GitHub repository show that each person in this team has been contributing to the simulation baseline.


Feb 27, 2022 – Apr 28, 2022
Contributions: Commits
Contributions to main, excluding merge commits and bot accounts
20
15
10
5
0
Feb 27
Mar 06
Mar 13
Mar 20
Mar 27
Apr 03
Apr 10
Apr 17
Apr 24
rbruflodt3
#1
43 commits 5,152 ++ 2,430 --
10
Feb 27
Mar 20
Apr 10
pcowhill3
#2
38 commits 4,908 ++ 3,588 --
10
5
Feb 27
Mar 20
Apr 10
jhouseknecht6
#3
38 commits 3,087 ++ 860 --
10
5
Feb 27
Mar 20
Apr 10



Below are descriptions of the significant contributions of each team member.

Rachel Bruflodt:

- Simulation architecture
- Default rules implementation
- Utility-based player modeling and implementation
- Long Route Player vs. Destination Ticket Player experimentation
- Other Player Agent experimentation

Patrick Cowhill:

- OOP components and program structure and layout within ICD
- Initial implementation of simulation game components
- MATLAB app expansion
- Alternative Ruleset exploration with treachery ruleset implementation
- Tie-breaker logic

Jessica Houseknecht:

- Train Card/Deck and Destination Ticket Card/Deck classes
- Initial random number generation
- Activity logging capability
- Metrics processing and comparison