AI-based Playtesting of Contemporary Board Games

Fernando de Mesentier Silva
New York University
Tandon School of Engineering
Brooklyn, New York 11201
fernandomsilva@nyu.edu

Scott Lee
New York University
Tandon School of Engineering
Brooklyn, New York 11201
sl3998@nyu.edu

Julian Togelius
New York University
Tandon School of Engineering
Brooklyn, New York 11201
julian@togelius.com

Andy Nealen
New York University
Tandon School of Engineering
Brooklyn, New York 11201
andy@nealen.net

ABSTRACT

Ticket to Ride is a popular contemporary board game for two to four players, featuring a number of expansions with additional maps and tweaks to the core game mechanics. In this paper, four different game-playing agents that embody different playing styles are defined and used to analyze *Ticket to Ride*. Different playing styles are shown to be effective depending on the map and rule variation, and also depending on how many players play the game. The performance profiles of the different agents can be used to characterize maps and identify the most similar maps in the space of playstyles. Further analysis of the automatically played games reveal which cities on the map are most desirable, and that the relative attractiveness of cities is remarkably consistent across numbers of players. Finally, the automated analysis also reveals two classes of failures states, where the agents find states which are not covered by the game rules; this is akin to finding bugs in the rules. We see the analysis performed here as a possible template for AI-based playtesting of contemporary board games.

CCS CONCEPTS

•Applied computing → Computers in other domains; Computer games; •Computing methodologies → *Artificial intelligence;*

KEYWORDS

Contemporary Board Games, Board Games, Artificial Intelligence, Playtesting, Ticket to Ride

ACM Reference format:
Fernando de Mesentier Silva, Scott Lee, Julian Togelius, and Andy Nealen. 2017. AI-based Playtesting of Contemporary Board Games. In *Proceedings of FDG'17, Hyannis, MA, USA, August 14-17, 2017,* 10 pages.
DOI: 10.1145/3102071.3102105

DOI: 10.1145/3102071.3102105

1 INTRODUCTION

As the popularity of board games has risen in recent years [13], so too has the speed with which they come to market [1, 3]. A substantial part of the board game design process is playtesting. Rules need many iterations and gameplay has to be balanced in order to guarantee a pleasant experience for the players of the final product. This work focuses on showing how artificial intelligence can be used to automate aspects of the playtesting process and the potential of such an approach, especially for Contemporary Board Game design.

With the the rejuvenation of board game design and popularity in the last few decades, a niche of tabletop games has been responsible for changing the paradigms of the industry. Its major features include but are not limited to: a Varying number of players (with many games going up to 4+), stochasticity, hidden information, multiple reinforcing feedback loops, underlying themes, various levels of player interaction and different degrees of strategic depth. Such games present an interesting challenge to many artificial intelligence techniques because of their incomplete information, randomness, large search space and branching factor. With excellent designs that result in minimal and elegant systems, appealing to an ever-growing player base, whether they are competitive or cooperative players, contemporary tabletop games are as relevant to the study of game design as video games.

During the process of development, designers experiment with rules to explore the space of possible actions and outcomes in their games. As their systems grow in complexity it becomes harder to control the scope of all possible scenarios that can develop from the different player interactions with the system as well as each other. Playtesting provides evidence for the reliability of the game as a system. A play session can reach edge cases that the designer didn't account for, which can also lead to observations about the many intricacies of the game. Under these scenarios, AI can be used to explore the game space and the consistency of the rules that govern the system. It can be used to search for unanticipated scenarios, fail states and potential exploits.

Playtesting also plays an essential role in designing a balanced experience, which is especially important in competitive multiplayer games. Game balance can take many forms: from ensuring that all players start with equal chances of winning to rewarding player's skills in-game. Competitive games in which players are rewarded for improving their skills have more success in retaining players [24]. In that regard it is usually desirable that players can climb a skill ladder [26] gradually as they experience the game. In contrast, having dominant or optimal strategies present in a game can be detrimental to its success. Players gravitate towards the dominant playstyles, abandoning other strategies. With the use of AI we can emulate powerful strategies under different scenarios in a feasible time frame. Being able to generate a large number of simulations enables us to assess undesirable outcomes and observe the impact that changes have on the gameplay, before having the game tested by human players.

1.1 Contributions of this paper

This paper presents an approach to analyzing board games using heuristic-based agents, and showcases this analysis method on the contemporary board game *Ticket to Ride*. We show that it is possible to both characterize the desirability of various parts of the map, the relative strengths of playing strategies in settings with different amount of players, the differences between maps in terms of what strategies work well for each map, and the effects of game mechanic changes between game variants. As a bonus, we also identify two failure cases, where the agents found game states that were not covered by the game rules.

The work presented in this paper expands on our previous workshop paper [11]. The previous paper was a pilot study using only two agents and presented our early findings to showcase a couple scenarios where it proved to show insight into the game design. The current paper uses four agents and analyzes eleven maps using two-, three- and four-player games, yielding a much deeper analysis.

2 BACKGROUND

Previous work has explored automating the process of balancing classic board and card games. Krucher used AI agents as playtesters of a collectible card game [25]. Based on the ratings assigned as a result of the agents' gameplay, an algorithm would automatically modify cards along different iterations. Hom et al. used a genetic algorithm to design a balanced abstract board game [17]. The algorithm searched the space of possible rules to alter the gameplay. Jaffe et al. used a educational perfect information game as the focus of a work on balancing [22]. In it the authors use metrics to weight the impact of multiple features in balancing the game. Dormans explores the economy of a game through the flow of its resources [12]. With it the author is capable of comparing multiple strategies in the early stages of the design. Meanwhile, Bayer et al. displays a model that encompasses both automatic and manual game balancing [2]. The authors utilize a prototype video game, Zombie Village Game, to compare results from both manual and automated game balancing. Our approach differs on the scope of the game being studied and how iterations over the design are proposed. Instead of automating the decisions regarding design changes, we propose using AIs to perform automated playtesting sessions, the results of which are presented to the designer in an effort to help facilitate informed decision-making regarding the next iteration of the game's design.

The large search space and stochasticity of Contemporary Board Games make them fairly complex by nature. This is further compounded by the fact that they allow for a varying number of players, which can facilitate a wide variety of interactions. While this may be rewarding or desirable for players, it presents a difficult challenge to designers wanting to thoroughly playtest their games. However, this burden can be ameliorated by leveraging AI driven playtesting, which allows designers to quickly and effortlessly modify the various conditions of their game and test the many permutations of their design space.

Contemporary Board Games have been the subject of some research. An AI framework was created by Guhe et al. for the game Settlers of Catan [14]. The authors used it to make agents tailored to certain strategies and compare its behavior to that of human players. Settlers of Catan was also explored by Szita et al. [38] and Chaslot et al. [8]. In those works MCTS agent performances were compared to that of heuristic based AIs. Meanwhile, Pfeiffer presented a reinforcement learning alternative that uses both learning and prior knowledge for playing game [31]. Heyden did work on the game Carcassone [15]. Focusing on the 2-player variant of the game, the author discusses strategies used for MCTS and Minimax search agents for playing the game. MCTS agents have also been explored for the game 7 Wonders by Robilliard et al. [32] and for the game Ticket to Ride by Huchler [18]. In Ticket to Ride the author has different agents playing against a cheating agent that has access to all hidden information, to compare their performances.

Discussing balance on a Contemporary Board Game has also been a subject of previous work. Mahlmann et al. tried to find a balanced card set for the game Dominion [28]. The work uses AI agents with different fitness functions and artificial neural networks (ANN) to evaluate the game state for the agents and game board. Results show that a specific cards were used by all different agents when they won matches. The authors proceed to conclude that these cards contributed to a balanced game regardless of play style. Following these findings, the authors demonstrate that the methods used can inform design and balance decisions in other games.

The idea of having a system that can provide assistance in the process of design is studied in the research field known as mixed initiative design [39]. The focus of this field is to study how can the computer act as a co-designer, contributing to the development process with inputs and suggestions. As far as the authors of this paper are aware, all work in this field has been done using video games. Liapis et al. created Sentient Sketchbook [27]. On it, users can create maps for Real Time Strategy games. The tool contributes by making suggestions on how the map can be altered to maximize specific objective functions. A stage editor for 2D platformer levels is presented by Smith et al. [37]. The systems gives the user the power to edit key aspects of the levels and then the rest of it is automatically filled by the system while also guaranteeing that it is playable. A tool for designing levels for the video game Cut the Rope was presented by Shaker et al. [34]. Using Ropossum, users can design their own levels and the system provides meanings as to test for playability, as well as the option to automatically generate a new level.

Other contributions have shown the ways in which AI can assist the game design process. Browne et al. used an evolutionary

Figure 1: The board for Ticket to Ride Europe.

algorithm to design games [6]. Utilizing measures of game quality [5], the algorithm successfully created a game that was later commercially published. Salge et al. took the concept of Relevant Information and use an AI to model and show its relation to game design [33]. Smith et al. presented a game engine capable of creating gameplay traces to explicitly state the behavior of the game [36]. Nelson presents seven alternative strategies to empirical playtesting to extract information from a game [29]. Nielsen et al. proposes characterizing the quality of a game in relation to the performance of multiple general game playing algorithms [30]. Isaksen explores the game space of the video game Flappy Bird using automated gameplay to encounter variants of the game that present relevant differences in game feel and other features [19, 20]. De Mesentier Silva et al. make use of AI and Machine Learning to generate simple and effective game playing heuristics for novices players in the game Blackjack [10].

3 TICKET TO RIDE

Ticket to Ride[9] is a 2-5 player competitive board game designed by Alan Moon and published by Days of Wonder in 2004. The game won multiple awards and sold over 3 million copies by 2014. Due to its success, multiple expansions and new versions of the game have been released since then. In this work we have used 11 different versions of the game across 6 different boards (USA, Europe, India, Nordic Countries, Netherlands and Asia) which feature different decks and small rule additions and modifications. Figure 1 shows the board for one of the most popular versions: Ticket to Ride Europe.

In Ticket to Ride players collect cards to claim train routes connecting different cities on the board. Players draw cards from a common pool, and cards can be of one of 9 different colors: Black, Blue, Green, Orange, Pink, Red, Yellow, White or Wild, with Wild a special type of card that can be used as if it were of any other color.

Once they have enough cards, they can claim a route between 2 cities. To claim any route, a player has to discard a number of cards of the same color from their hand. The color and quantity of cards are determined by the route they are trying to claim. After discarding their cards, players proceed to placing their train tokens on the route to mark as theirs. A route claimed by one player can no longer be claimed by any other. The game reaches an end when one player is left with 2 or less trains at the end of a turn, at which point all players proceed to take one more turn. The game then ends and all players total up their scores. The player with the highest score wins.

Players score points by claiming routes or by completing their Destination Cards. A route's size determines how many points it awards. Additionally, Destination Cards are drawn at the beginning of the game and more can be drawn during play. Each card defines two cities that the player can connect by claiming routes. If a player is able to connect these two cities, they will be awarded extra points in the end of the game. However, if a player fails to connect the two cities on any of her Destination Cards, they get a penalty of the same amount that they would be rewarded if they were successful. Each card has the amount of point it awards/penalizes written on it.

Ticket to Ride is considered a gateway game, a game used to introduce new players to Contemporary Board Games, and a family game, a game that is meant to be played by players of any age. It is regarded highly among the Contemporary Board Game community for its simple rules and space for developing different levels of strategy.

4 AGENTS

Ticket to Ride has several elements that make it challenging for many artificial intelligence techniques. The game has a large space of possible configurations. For a 2 player game the number of possible game states is of the order of 10^{54} [18]. Additionally, stochasticity is present in the deck of Train Cards and on the deck of Destination Cards. A player's hand of cards is also kept hidden from other players. It is therefore difficult to formulate a heuristic function to analyze the state of the board. Due to these challenges we were unable to implement effective agents that use techniques such as A-Star and MCTS.

With our lack of success with search based AIs we decided to create agents tailor made for Ticket to Ride. Due to the commercial success of the game, there is a strong community that actively plays and discusses it. Guided in part by our experience of playing the game for several years and by researching the community discussions of game strategies [4] we arrived at 4 agents with distinct playstyles. With that being said, we do believe that it is possible to achieve a better playing MCTS agent with some modifications, following a similar approach done by Jacobson et al. for Super Mario Bros [21] or by Huchler for Ticket to Ride [18]. Alternatively, it may be possible to use an evolutionary approach as seen for other games with high branching factors [23].

Another guiding point for this work is trying to emulate ideal conditions of playtesting. In this sense there is an argument to be made against having very strong agents, such as DeepBlue [7] in Chess and AlphaGo [35] in Go, that are above human level. Playtesting many times is aimed at modeling human play sessions. For such, it is interesting to have players that are not completely new to the game, but that also don't excel at the game. Agents that play at the same level as an average player would be more useful in approximating its target audience.

4.1 Destination Hungry Agent (DHA)

Destination Hungry Agent tries to emulate a playstyle centered around executing a long term strategy. It is a strategy that attempts to formulate a plan at the beginning that maximizes potential points and focuses on executing it over the course of the game, only making adjustments if necessary.

This agent's focuses on scoring points by accumulating Destination Cards at the beginning of the game. It will start the game by obtaining Destination Cards until it reaches a threshold related to the number of trains trains required to complete the drawn destinations. It will choose to keep the Destination Cards that maximize the number of points scored per Train Token needed to connect the cities. This strategy heavily favors Destination Cards that can be completed by using routes that other Destination Cards would also use.

The agent will then proceed to generate a list of routes it needs to connect any city in any of its Destination Cards to any other city in its Destination Cards. This list is recalculated any time a player claims a route in the game, to ensure that the routes it plans to have at the end of the game are still available.

After the agent arrives at the list of routes it needs to have at the end game it evaluates which Train Cards it would need to claim those routes. It will then build a priority list of the Train Cards based on how many of each it requires. This list is updated every time it manages to claim a route. The list decides which Train Cards the agent will attempt to obtain.

To choose an action for a given turn, the agent will prioritize drawing Destination Cards. Once it reaches its threshold of Destinations, it will attempt to claim a route it needs. Otherwise it will draw Train Cards. In the event it has claimed all routes it desires, the agent will aggressively claim other routes in order to spend Train Tokens and end the game as quickly as possible.

4.2 Route Focused Agent (RFA)

Route Focused Agent tries to emulate one of the most common strategies in the game. The agent attempts to score points using a fairly straightforward approach: construct toward the obvious long-term objectives, completing the Destination Cards, then perform the individual actions with the highest rewards, claiming the longer routes.

This agent start uses only the Destination Cards it receives at the setup of the game. It never decides to draw any new Destination Cards. This strategy involves completing Destination Cards quickly while prioritizing long routes to score extra points.

On a given turn it checks which Destination Cards have not been completed. It then proceeds to evaluate what other it needs to complete them. All routes it deems necessary get added to a priority queue. The score of a route is the sum of the route's point value and the point value of the Destination Card it contributes to multiplied by 2. The Destination points are multiplied by 2 to reflect the fact that it will get penalized if it doesn't complete them. After that it evaluates every other route still available in the game and proceeds to add those to the priority queue using only its point value as its score.

Every turn the agent selects an action by first looking if it can claim any of the top priority routes in the queue. If it can, it will proceed with that move. If it cannot, it will decide to draw Train Cards based on the color of the highest priority route in the queue.

4.3 One Step Agent (OSA)

This careful/conservative agent makes decisions one turn at a time. It decides on what it believes to be the next most important step to take and it then builds to achieve it. It attempts to complete one Destination at a time, one route at a time. Once it has finished with its cards, it draws new ones, unless the game is close to its end.

The One Step Agent starts the game selecting the Destination Cards it wishes to keep based on the points of points per train token it awards, in a similar fashion to what is done by DHA. During the game, the agent will only decide to draw new Destination Cards once it has been able to complete all the ones it currently owns.

Every turn it checks the Destination Cards it owns to see which have not yet been completed. From this set, it prioritizes completing the cards that awards more points. It then identifies the routes required to complete this card by calculating the shortest path between the cities. From the list of unclaimed routes for that Destination, it prioritizes the least expensive routes. A route's cost is defined by the number of cards required to claim it, and is also dependent on the cards the agent is currently holding.

Once it has decided on a route, it will check to see if the route can be claimed. If it cannot, it will draw cards of the color of that route. Once all Destination Cards it owns are completed, it will choose to draw new Destination Cards, unless it has less than 5 Train Tokens. In this case it will find the largest route it can claim with the number of trains it has left.

4.4 Long Route Agent (LRA)

The Long Route Agent represents a somewhat unorthodox strategy. Its playstyle favors longer routes on the board. It is an effort to both profit from the amount of points they award and capture key routes before other players. Many times the points it could achieve from Destination Cards end up being sacrificed to facilitate this strategy.

This agent's strategy is derived from a claim put forth by members of the game's playerbase which asserts that completing Destination Cards is not a requirement to win at the game [4]. The agent therefore attempts to win while ignoring routes of size 1 and 2 on the board, which have generally low value apart from completing Destination Cards.

The agent selects which Destination Cards to keep using the same strategy as that of the One Step Agent, with a notable difference in that the route planning phase ignores routes that take less than 3 trains to claim. After choosing which cards to keep at the beginning it will never draw new Destinations.

To select which move to make it looks at all routes it requires to complete its Destinations. It will proceed to evaluate all routes it wants to claim by the end of the game, giving priority to longer routes, routes that can be taken with any color (routes painted gray on the board) and routes that requires the fewest additional draws (already have some of the cards necessary in hand). If it already succeeded in completing all its Destinations, it select a route using the same criteria, but will now consider all unclaimed routes on the board that are of size 3 or more.

Every turn, if it can claim the top priority route it will make that move. If not, it will proceed to draw Train Cards that match the color of such route.

5 ANALYSIS OF THE GAME

Experiments were run with the 4 agents detailed in the previous section for 11 variants of the game, all which have been commercially released. For each variant, we simulated all possible matchups between different agents for 2, 3 and 4 player games. For each of these setups 1000 matches were played, summing up 10.000 matches per variant for a total of 110.000 games overall.

All variants of the game were implemented in full, with all their specific rules, except for the Netherlands variant. The Netherlands game was not run using its unique money management rules, but instead used the variant proposed in the game's rulebook of using the same rules as for the original USA version.

In this section we highlight interesting data points and visualizations to show how the designer could benefit from AI driven playtesting. We showcase both general analysis, that could be used for most games, as well as observations specific to Ticket to Ride. Our belief is that the results are flexible enough to create approaches that allow for the exploration of a variety of games.

5.1 Agent Matchup

Figure 2 shows the result of the matches played in all variations. For each variation listed, two charts are being shown: the left chart shows results for 4-player games and the right chart for 2-player games. The 2-player aggregation represents the average win ratio of each agent for each of the 2-player match-up possibilities, one against each of the other agents.

The charts point to some very interesting behaviors related to the agents. First, agent performance changes considerably from map to map, which indicates that different strategies have varying levels of efficacy depending on which variant of the game is being played. In terms of game design, it points to evidence that the changes in gameplay were meaningful between the game variants.

Another noticeable pattern is that the same agent on the same variant many times performs differently in a 4-player game compared to to a 2-player game. The results for the India variant shown on Figure 2h highlights this behavior. For example, the LRA agent has a 49.4% win ratio in 4-player games, being the dominant player in that scenario, while it performs substantially worse in 2-player games, with a 27.3% win rate compared to the 30.7% of OSA.

To further understand the differences in results when varying the number of players, we show more detailed statistics for the India variant in Table 1. The table details the agents' strategies using a number of measures: their average point score, how successful they are in completing their Destination Cards, how much they profit (in points) from Destinations on average and how many Train Tokens they have left at the end of the game.

The clustered nature of the India game board encourages players to interfere with one another's strategies. It also happens that most of the longer routes are present on the outer part of the map, rather than in the middle. As such, in 2-player games, LRA is less likely to interfere with the other agents' routes, since it is more likely to claim outer routes. Meanwhile, in 4-player games the other 3 agents are building on the inner routes, interfering with one another, while LRA builds on the outer routes with minimal obstruction. That is reflected on the average scoring, LRA is the lowest scoring agent in 2-player games, while being the highest scoring in 4-player.

The control and freedom of LRA on 4-player games is also evident when observing the average Leftover Trains. It is common for the player that triggers the end game, by having 2 or less Train Tokens left, to have an edge over the other players. Triggering the end game usually results in other players not being able to complete Destination Cards. It also leaves players with underutilized train tokens, which normally would be converted into score by claiming routes. Observing the average Leftover Trains, it is very clear that LRA was triggering the end of the game on most matches, while the other agents were left with an average of 9 trains more at the end.

When looking at the average number of points scored with Destination Cards, one can see the differences between the strategies of the agents. While LRA is frequently losing points from its destinations, having long routes as its main source of income, OSA scores at least 25% of its points from Destinations .

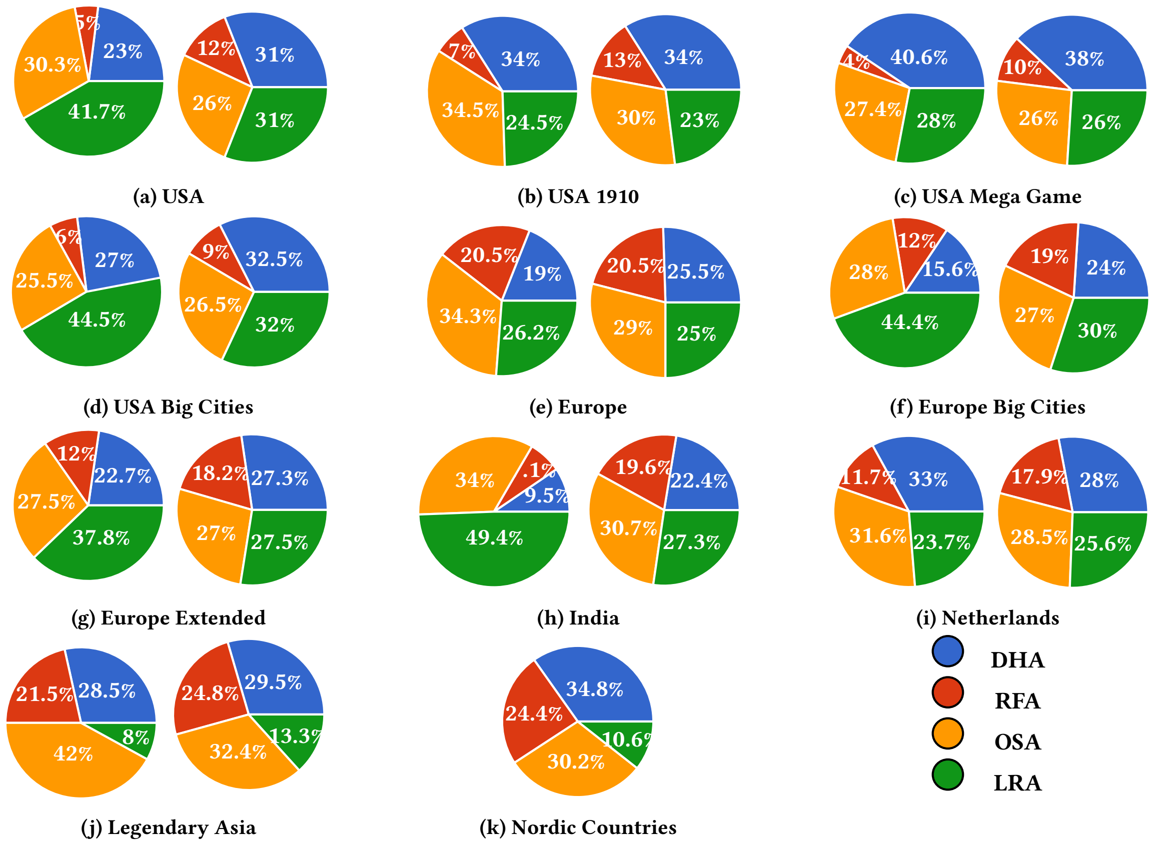


Figure 2: Agent Matchup results for each map with 4-player on the left and 2-player aggregation on the right. Nordic Countries is an exception since the map is for up to 3 players, so only the chart for 2-player aggregation is being shown.

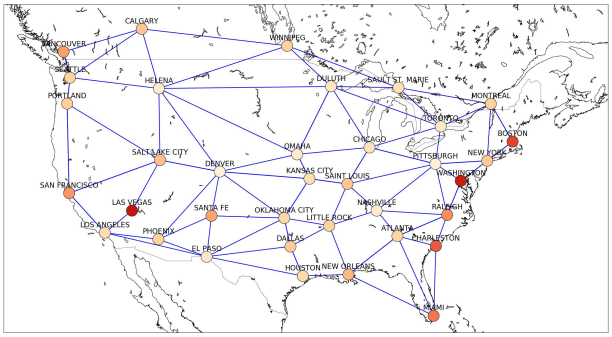
Game Mode	Agent	Win Ratio	Avg. Score	Avg. Dest. Completed	Avg. Dest. Points	Avg. Leftover Trains
2 players (AGG)	**DHA**	0.224	81.81	0.90	32.11	6.8
	RFA	0.196	80.38	0.84	17.29	2.3
	OSA	0.307	94.08	0.84	26.44	5.3
	LRA	0.273	75.48	0.60	11.61	6.7
4 players	**DHA**	0.095	42.75	0.66	6.63	13.7
	RFA	0.071	45.94	0.58	2.12	10.5
	OSA	0.34	76.13	0.74	19.55	9.4
	LRA	0.49	84.49	0.16	-8.61	0.7

Table 1: Detailed data from the India Map. The table shows 2-player aggregated games and 4-player games. Destinations completed is the percentage of Destination Cards it completes in average. Destination Points is the average of points the agent scores from Destination Cards (including points lost for not completing them). Leftover Trains shows the average amount of trains the agent has left at the end of the game. This table highlights the disparity between agents performances in 2-player games and 4-player games on this map.

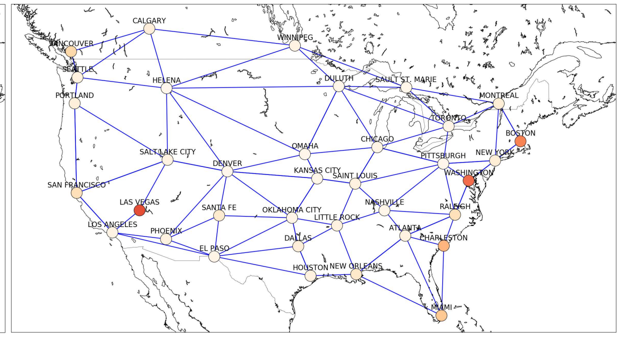
5.2 Scenarios not covered by the game rules

When playing the game, our Agents encountered scenarios that were not covered by the written rules of the game. The two cases found created situations wherein at least one of the players was left with no moves to make. The scenarios were edge cases in the game, meaning they are unlikely to happen on regular play sessions. That said, searching and exploring the action space of the game for gamestates that are unresolved can be a valuable asset for games during the process of design.

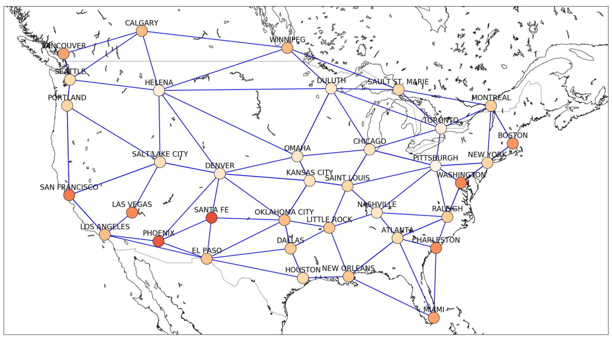
No Train Cards This scenario is consequence of one player repetitively drawing Train Cards without actually claiming any routes. The scenario only happens if the other players are not capable of ending the game before the deck of Train Cards is depleted.



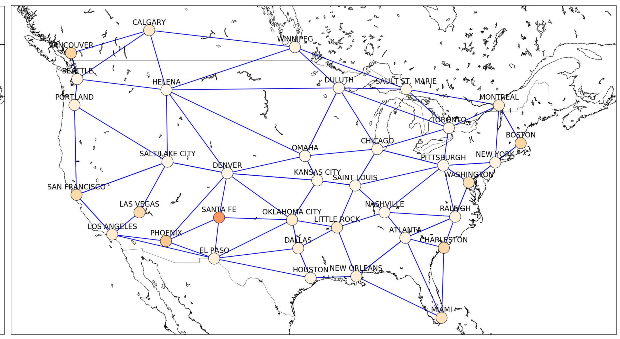
(a) USA 2-player Aggregate



(b) USA 4-player



(c) USA 1910 2-player Aggregate



(d) USA 1910 4-player

Figure 3: Color map showing how often any routes to a specific city are claimed. The more red a city is on the map, the more undesirable it is. Undesirability is a factor of the number of games simulated where not a single agent claimed any routes connecting that city. The figure shows the board for USA and USA 1910 for 2-player and 4-player games.

Once there are no more Train Cards to draw and a player does not have enough cards to claim any route in the board, the only move left is to draw more Destination Cards, which will likely force them to lose points at the end. Once the Destination deck is also depleted, the player is left without any legal moves to take in is turn.

3 Wild Loop This scenario is caused by two rules chain reaction. The first rule states that when 3 out of the 5 face up Train Cards happen to be wild cards, all face up cards are discarded, and 5 new cards are added from the deck. The second rule tells players that once the Train Card deck is depleted, the discard pile is to be shuffled to form the new deck. If players exclusively draw non-wild Train Cards, it is likely that the deck will become short and filled with wilds. At some point, 3 or more wild cards will be part of the face up set, which will trigger a discard and replenish. However, if there are no more cards in the Train deck, the discard will be reshuffled, which would cause the wild cards to return to the face up set, which would once again trigger the reshuffle. This chain reaction would happen indefinitely, without any players being able to intervene, leaving the game stuck in an infinite loop.

5.3 Undesirable Cities

Figure 3 shows how frequently cities are claimed by agents on the USA board. Red coloring reflects a city that is undesirable to players. The figure shows a comparison for two variants played on the same board, USA and USA 1910. The only difference between these two variants is the deck of Destination Cards, each variant has their own. Results are shown for 2-player and 4-player games.

The figure demonstrates the impact of changing even a single component in a game. By changing the deck of Destinations, the distribution of claimed routes has changed. Las Vegas and Washington are the two most undesirable cities in the USA variant, for both 2 and 4 player games, were replaced by Santa Fe as the most undesirable in USA 1910. That is explained by the fact that no Destination Cards in USA had Las Vegas or Washington as targets, meanwhile each were present in 3 cards for USA 1910. Santa Fe on the other hand went from being present in 2 cards to being in 1.

Another point to be noticed is when comparing the two 4-player maps. While Las Vegas, Washington and, to a point, Boston stand out as undesirable in USA, only Santa Fe, and in a more modest way, stands out in USA 1910. So as a consequence of changing the cards, gameplay is more spread out on the board, with more cities appearing to have similar value in terms of strategy.

To quantitatively verify that the relative ranking undesirability of cities changes little with the number of players but much more with the destination cards, we calculated the Pearson correlation between the undesirability of cities in the four map conditions. For the USA map, the correlation between 2-player and 4-player mode was 0.94; for the USA 1910 map, the correlation between 2-player and 4-player mode was 0.92. In other words, undesirability in 2-player mode is a very strong predictor of undesirability in 4-player mode and vice versa. On the other hand, the correlation between

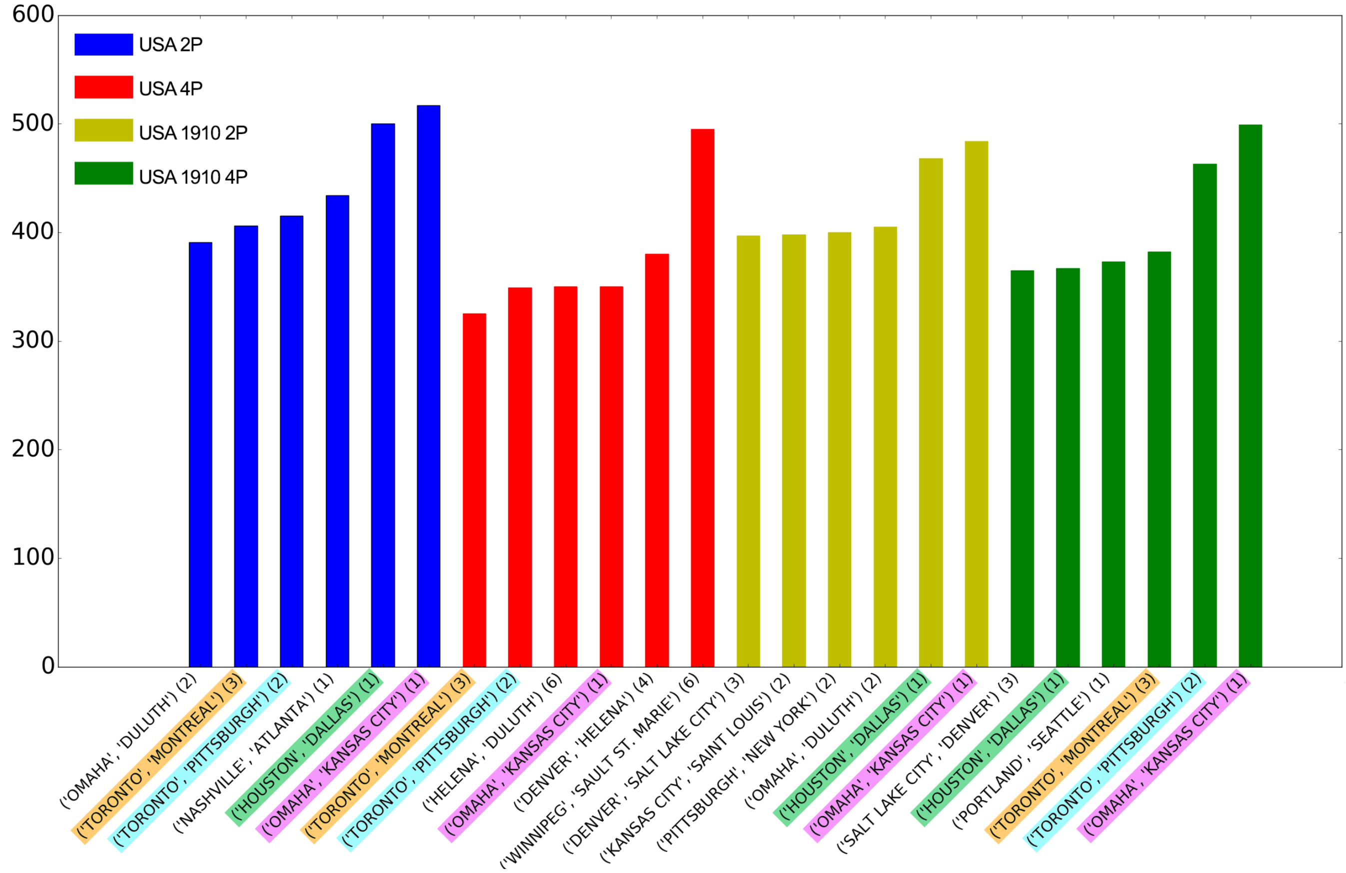


Figure 4: Bar Chart showing the top 6 routes claimed by the winners, for USA and USA 1910 variants, for both 2-player and 4-player games. The x-axis display the names of the routes, represented by the 2 cities it connects, with its size in parenthesis. The y-axis shows how many games, out the 1000 simulations, the winning player claimed that route.

undesirability of cities in the USA and USA 1910 map was only 0.71 for the 2-player mode, and for the 4-player mode the correlation drops to 0.52. Changing the destination card thus has a very strong effect on the relative desirability of cities.

5.4 Most Useful Routes

Figure 4 shows the routes that were most claimed by winners on the USA and USA 1910 variants. The top 6 routes are displayed for both 2-player and 4-player games in both variants. Highlighted are the names of routes that are common across at least 3 of the 4 scenarios.

The figure helps highlight routes that are apparently crucial for winning strategies on the USA game board, with 4 routes, Toronto-Montreal, Toronto-Pittsburgh, Houston-Dallas and Omaha-Kansas City, being predominant. Toronto stands out as a city for being present in 2 of those 4 routes, which is explained by the fact the it is one of the most connected cities in the map, being part of 5 routes and also being a strong step in connecting the cities on the West Coast to New York.

Out of the 4 routes highlighted, only Toronto-Montreal has size more than 2 (size 2 is the most common in the board, and size 1 the easiest to claim). This also indicates this route as being key for playing strategies that aim to complete East to West Destination Cards (the most valuable in the game) using the north side of the map. The bar chart also validates the community claims that Houston-Dallas is an important choke point and that Toronto-Pittsburgh is a strong route and should be claimed more often then not[4].

6 DISCUSSION AND FUTURE WORK

We believe that this work can be extended to inform a more robust system capable of fine tuning the different variables of the game. The work shown on this paper can be used to act as an evaluation function for an evolutionary algorithm, enabling it to explore different variants for the game. With assistance from the designer, an evolutionary technique could change or produce elements of the game, such as board elements or card decks. Construction could be guided to change the board to fit certain playstyles, or to guarantee that different strategies are similarly effective. Even more, it could be used as a guide to a procedural generation algorithm to create new content.

This work has also the potential to explore variations of the game tailored for specific design goals. A designer after analyzing the results from playtesting could decide that it is desirable to explore how constraints can be adjusted to reduce the number of turns in a game. The system could then emulate the impact of changing the number of cards drawn every turn or the number of trains with which the player starts. Other possible goals could be to encourage players to score more points on average or to interact more often.

The results shown in this paper lead us to believe that this technique can be most profitable in the early stages of design. One future work we are interested in exploring is using these findings for a game being designed as opposed to a game that has already been released. Of course that poses challenges of a different nature, one major decision involves selecting AI techniques for a game of which we have yet to obtain substantial knowledge. Potential approaches that require minimal domain knowledge include reinforcement learning, heuristic generation [10], procedural personas [16] and evolutionary techniques.

7 CONCLUSIONS

In this paper, we have presented a method for AI-based analysis of board games and applied it to the popular contemporary board game *Ticket to Ride*. The four heuristic agents used, *Destination Hungry*, *Route Focused*, *One Step*, and *Long Route*, represent different strategies that can be taken when playing the game, based on our own analysis of strategies for playing the game. All of the agents were shown to be effective on some maps, with the three of the agents being the *most* effective in at least one configuration. The analysis of the game showed that the maps varied in what strategies worked best, and the number of players also had a significant effect on the relative performance of the strategies. The relative performance profiles can be seen as a fingerprint of each map, and allows us to see e.g. that Europe Extended is strategy-wise similar to the original USA map, whereas India is dissimilar to all other maps, especially in two-player mode. Analyzing the most desirable cities across maps, we find a very stable pattern across numbers of players, but the desirable cities instead change markedly when different destination cards are used for the same map. The list of most useful routes on the other hand changes both when different destination cards are used on the same map and when different numbers of players play use the same map with the same destination cards. Interestingly, in the process of analyzing the game we could also identify two types of states which "break the game" in the sense that they are not covered by the game. We believe that automated analysis of contemporary board games with automated agents has a lot to offer, as exemplified by the results in this paper.

ACKNOWLEDGMENTS

Authors thank the support of CAPES, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil.

REFERENCES

[1] Marcus Beard. 2016. *A Data Analysis of Board Game Rankings*. http://www.bestplay.co/board-games-getting-worse/ Accessed: 2017-02-01.
[2] Marlene Beyer, Aleksandr Agureikin, Alexander Anokhin, Christoph Laenger, Felix Nolte, Jonas Winterberg, Marcel Renka, Martin Rieger, Nicolas Pflanzl, Mike Preuss, and others. 2016. An integrated process for game balancing. In *Computational Intelligence and Games (CIG), 2016 IEEE Conference on*. IEEE, 1–8.
[3] Thomas Bidaux. 2017. *Kickstarter in 2016 fi Deep dive into the Games category*. http://icopartners.com/2017/01/kickstarter-2016-deep-dive-games-category/ Accessed: 2017-03-01.
[4] Board Game Geek. 2006. *Ticket to Ride Strategy - A Complete Strategic overview*. http://boardgamegeek.com/thread/142743/strategy-complete-strategic-overview Accessed: 2017-02-01.
[5] Cameron Browne. 2008. *Automatic generation and evaluation of recombination games*. Ph.D. Dissertation. Queensland University of Technology.
[6] Cameron Browne and Frederic Maire. 2010. Evolutionary game design. *IEEE Transactions on Computational Intelligence and AI in Games* 2, 1 (2010), 1–16.
[7] Murray Campbell, A.Joseph Hoane, and Feng hsiung Hsu. 2002. Deep Blue. *Artificial Intelligence* 34, 1 (2002), 57 – 83. http://www.sciencedirect.com/science/article/pii/S0004370201001291
[8] Guillaume Chaslot, Sander Bakkes, Istvan Szita, and Pieter Spronck. 2008. Monte-Carlo Tree Search: A New Framework for Game AI.. In *AIIDE*.
[9] Days of Wonder. 2004. *Ticket to Ride*. https://en.wikipedia.org/wiki/Ticket_to_Ride_(board_game) Accessed: 2016-05-15.
[10] Fernando de Mesentier Silva, Aaron Isaksen, Julian Togelius, and Andy Nealen. 2016. Generating Heuristics for Novice Players. *2016 IEEE Conference on Computational Intelligence and Games* (2016).
[11] Fernando de Mesentier Silva, Scott Lee, Julian Togelius, and Andy Nealen. 2017. AI as Evaluator: Search Driven Playtesting of Modern Board Games. (2017).
[12] Joris Dormans. 2011. Simulating Mechanics to Study Emergence in Games. *Artificial Intelligence in the Game Design Process* 2, 6.2 (2011), 5–2.
[13] Milton Griepp. 2016. *Hobby Games Market Nearly 1.2 Billion in 2015*. http://icv2.com/articles/news/view/35150/hobby-games-market-nearly-1-2-billion Accessed: 2016-10-29.
[14] Markus Guhe and Alex Lascarides. 2014. Game strategies for the Settlers of Catan. In *2014 IEEE Conference on Computational Intelligence and Games*. IEEE, 1–8.
[15] Cathleen Heyden. 2009. *Implementing a computer player for Carcassonne*. Ph.D. Dissertation. Maastricht University.
[16] Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N Yannakakis. 2014. Evolving personas for player decision modeling. In *Computational Intelligence and Games (CIG), 2014 IEEE Conference on*. IEEE, 1–8.
[17] Vincent Hom and Joe Marks. 2007. Automatic design of balanced board games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*. 25–30.
[18] Carina Huchler. 2015. *An MCTS agent for Ticket to Ride*. Master's Thesis. Maastricht University.
[19] Aaron Isaksen, Dan Gopstein, and Andy Nealen. 2015. Exploring game space using survival analysis. *Foundations of Digital Games* (2015).
[20] Aaron Isaksen, Dan Gopstein, Julian Togelius, and Andy Nealen. 2015. Discovering unique game variants. In *Computational Creativity and Games Workshop at the 2015 International Conference on Computational Creativity*.
[21] Emil Juul Jacobsen, Rasmus Greve, and Julian Togelius. 2014. Monte mario: platforming with MCTS. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. ACM, 293–300.
[22] Alexander Jaffe, Alex Miller, Erik Andersen, Yun-En Liu, Anna Karlin, and Zoran Popovic. 2012. Evaluating Competitive Game Balance with Restricted Play.. In *AIIDE*.
[23] Niels Justesen, Tobias Mahlmann, and Julian Togelius. 2016. Online Evolution for Multi-action Adversarial Games. In *European Conference on the Applications of Evolutionary Computation*. Springer, 590–603.
[24] Eva Kraaijenbrink, Frank van Gils, Quan Cheng, Robert van Herk, and Elise van den Hoven. 2009. Balancing skills to optimize fun in interactive board games. In *Human-Computer Interaction–INTERACT 2009*. Springer, 301–313.
[25] Jonas Krucher. 2015. *Algorithmically Balancing a Collectible Card Game*. Bachelor's Thesis. ETH Zurich.
[26] Frank Lantz, Aaron Isaksen, Alexander Jaffe, Andy Nealen, and Julian Togelius. 2017. Depth in Strategic Games. (2017).
[27] Antonios Liapis, Georgios N Yannakakis, and Julian Togelius. 2013. Sentient Sketchbook: Computer-aided game level authoring.. In *FDG*. 213–220.
[28] Tobias Mahlmann, Julian Togelius, and Georgios N Yannakakis. 2012. Evolving card sets towards balancing dominion. In *Evolutionary Computation (CEC), 2012 IEEE Congress on*. IEEE, 1–8.
[29] Mark J. Nelson. 2011. Game metrics without players: Strategies for understanding game artifacts. In *Proceedings of the First Workshop on AI in the Game-Design Process*. 14–18.
[30] Thorbjørn S Nielsen, Gabriella AB Barros, Julian Togelius, and Mark J Nelson. 2015. General video game evaluation using relative algorithm performance profiles. In *European Conference on the Applications of Evolutionary Computation*. Springer, 369–380.
[31] Michael Pfeiffer. 2004. Reinforcement learning of strategies for Settlers of Catan. In *Proceedings of the International Conference on Computer Games: Artificial Intelligence, Design and Education*.

[32] Denis Robilliard, Cyril Fonlupt, and Fabien Teytaud. 2014. Monte-carlo tree search for the game of fi7 wondersfi. In *Computer Games*. Springer, 64–77.
[33] Christoph Salge and Tobias Mahlmann. 2010. Relevant information as a formalised approach to evaluate game mechanics. In *Computational Intelligence and Games (CIG), 2010 IEEE Symposium on*. IEEE, 281–288.
[34] Noor Shaker, Mohammad Shaker, and Julian Togelius. 2013. Ropossum: An Authoring Tool for Designing, Optimizing and Solving Cut the Rope Levels.. In *AIIDE*.
[35] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, and others. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529, 7587 (2016), 484–489.
[36] Adam M Smith, Mark J Nelson, and Michael Mateas. 2010. Ludocore: A logical game engine for modeling videogames. In *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games*. IEEE, 91–98.
[37] Gillian Smith, Jim Whitehead, and Michael Mateas. 2010. Tanagra: A mixed-initiative level design tool. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games*. ACM, 209–216.
[38] István Szita, Guillaume Chaslot, and Pieter Spronck. 2009. Monte-carlo tree search in settlers of catan. In *Advances in Computer Games*. Springer, 21–32.
[39] Georgios N Yannakakis, Antonios Liapis, and Constantine Alexopoulos. 2014. Mixed-initiative co-creativity. In *Proceedings of the 9th Conference on the Foundations of Digital Games*.