

# Game Playing AI for Settlers of Catan

David Martin

Supervisors: Dr Lars Kunze and Dr Nick Hawes



Submitted in conformity with the requirements for the  
degree of BSc Computer Science School of Computer  
Science University of Birmingham

School of Computer Science  
University of Birmingham



# Abstract

## Game Playing AI for Settlers of Catan

David Martin

---

Game Playing is both a useful and important way of measuring the performance of artificial intelligence techniques. Commonly these techniques are tested on games such as Chess, Backgammon or more recently, Go. While these games are very well researched and explored there are many other games that have not had as much attention yet still provide very interesting characteristics and features that make them worthy of exploration.

In this project, we design and implement a “bot” that is capable of playing the popular board game Catan, also popularly known as Settlers of Catan. This game is interesting as a variety of techniques must be employed in order to

By building on past work on AI for this game a system is developed that relies on Monte Carlo Tree Search and applied heuristics in order to play the game to a strong level given enough time to calculate moves.

A set of tests are then run against a baseline AI showing that our bot performs well when tested over a number of games and is capable of good play against an average human player and respectable play against an expert one, although it remains evident that there are obvious improvements that could be made to our system which would further boost its performance.

# Acknowledgments

---

I would like to thank my supervisors for this project Dr Lars Kunze and Dr Nick Hawes for their advice and guidance without which this project would not have been possible.

Additionally, I would also like to thank Jeremy Monin, the maintainer and lead developer of the open source JSettlers2 program, which provides the engine for my project for providing me with guidance on the usage of the program and for incorporating helpful changes into the software. Without this software the project would not have advanced as far.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Game Playing . . . . .	1
1.2	Related Work . . . . .	1
1.3	Aims . . . . .	2
<b>2</b>	<b>Background</b>	<b>3</b>
2.1	Catan . . . . .	3
2.2	Rules of the Game . . . . .	3
2.2.1	The Board and Playing Equipment . . . . .	3
2.2.2	Beginning the Game . . . . .	4
2.2.3	Regular Turn . . . . .	4
2.2.4	Building . . . . .	4
2.2.5	The Robber . . . . .	5
2.2.6	Development Cards . . . . .	5
2.2.7	Trading . . . . .	5
2.2.8	Winning the Game . . . . .	6
2.3	Catan Strategy . . . . .	6
2.3.1	General Principles . . . . .	6
2.3.2	Common Human Strategies . . . . .	7
2.3.3	Opening Turn . . . . .	7
2.3.4	Robber Strategy . . . . .	7
2.3.5	Trading . . . . .	8
2.4	Common AI Techniques for Game Playing . . . . .	8
2.4.1	Minimax Search . . . . .	8
2.4.2	Monte Carlo Tree Search . . . . .	9
2.4.3	Expert Systems . . . . .	9
2.4.4	Reinforcement Learning . . . . .	9
2.4.5	Evolutionary Algorithms . . . . .	9
2.5	JSettlers2 . . . . .	9
2.5.1	JSettlers2 Structure . . . . .	9
<b>3</b>	<b>Implementation and Development</b>	<b>10</b>
3.1	Overview and Plan . . . . .	10
3.2	Interaction with API . . . . .	10
3.3	Opening move generator . . . . .	10
3.4	Move generator . . . . .	10
3.5	Monte Carlo Tree Search . . . . .	10
3.5.1	Handling Non-Determinism . . . . .	10
3.5.2	UCT Formula . . . . .	10
3.5.3	Application of expert knowledge . . . . .	10
3.6	Simulator . . . . .	10
3.6.1	Simulator Heuristics . . . . .	10
3.6.2	Simulator Fidelity . . . . .	10
3.6.3	Simulator Performance . . . . .	10
3.7	Optimisation . . . . .	10
<b>4</b>	<b>Results</b>	<b>11</b>
4.1	Effects of Heuristic Opening Strategy . . . . .	11
4.2	Effects of Adjustment of Simulation Count . . . . .	11
4.3	Effects of Adjustment of UTC Exploration Parameter . . . . .	11
4.4	Play against Humans . . . . .	11
4.5	Qualitative analysis of Play . . . . .	11

**5 Evaluation 12**

5.1 Performance . . . . . 12

5.2 Comparison with Other Systems . . . . . 12

5.3 Limitations and Issues . . . . . 12

**6 Conclusion 13**

6.1 Possible Future Work . . . . . 13

# 1 Introduction

## 1.1 Game Playing

While at first the idea of game playing may seem irrelevant to the development of AI, upon further inspection it transpires that work on this area not only helps to progress the field of artificial intelligence but it also provides a way in which various techniques can be effectively tested and evaluated in a controlled environment.

Board games are most commonly used for this area of research as they can generally be easily represented on a computer and their rules can be changed and the environment modified. Several different games are commonly used for research into game playing AI thanks to the different features that each game possesses. Perhaps the most ubiquitous game in this field is chess which has had significant effort spent since at least the late 1950's. Chess provides a good choice for the development of basic game AI as it is a game of perfect information, contains only one opposition player and has a very rich source of expertise surrounding the game while at the same time being complex and with a large enough branching factor that the game could not be solved using brute force methods.

Backgammon is also commonly used in the field of game playing AI. It shares most of the features that chess has but at the same time introduces an element of chance through the usage of dice. This naturally adds further complications to the techniques used to play the game.

Despite the relative complexities of these games AI has been developed to play both backgammon and chess beyond the level of the best human players. The best chess bots have been unable to be beaten by humans for a number of years now and backgammon theory was significantly advanced by the development of AI which played the game in methods previously unseen by human players.

More recently in this area focus has shifted to applying new techniques to games that had previously not seen top level humans beaten by robots on a consistent basis such as go and poker.

On top of its relevance to the development of game playing artificial intelligence also has strong applications in other fields such as game theory, which provides the mathematical basis behind much of the research in this area, and economics. In an economic scenario economies can be modelled as games with the different agents in the system competing and cooperating in order to maximise their own gain.

In this project we will design and implement an AI that can play the popular board game Catan. This game features elements that are considerably more complex than both chess and backgammon. The number of moves that can be played on any given turn is generally larger and like backgammon there are dice rolls which introduce non-determinism into the game. Further complications are added by the multi-player nature of the game as 4 people play it at once, imperfect information, randomisation of the board and player-to-player trading. All aspects of a game that are not issues in Chess and Backgammon. While bots exist for Chess, Backgammon and recently Go and Poker that can beat the best human players, to our knowledge, there does not exist a bot that can yet consistently defeat the best Catan players in the world.

## 1.2 Related Work

In comparison to other more recently fashionable games Catan has received only a small amount of attention from researchers. We imagine that this is mostly due to the fact that rule set of the game is considerably more complex than that of other games such as Chess and Go. In these games while the strategy of playing the game may be equally as hard the rules are generally very simple to implement on a computer, with chess for example being able to be represented in a mere 455 bytes of data and go being able to be represented in an amount that is not considerably higher.

Despite the relative complexity of the rules of Catan there do exist open source implementations of the game which can be used to run the game engine and facilitate the design of bots that can play the game. Some research into the game does exist however. Szita et al. (2009) describes how a game of Catan can be played by utilising Monte Carlo Tree Search (MCTS). This paper shows that it is possible for MCTS to produce good results against bots using more naive, heuristic only approaches. This paper does not however provide much depth into the strategy employed by the MCTS, neglecting to mention how moves were generated for the MCTS and the details of the implementation of the MCTS are somewhat vague with only small details given about how the problem of non-determinism is tackled within the tree search. There are also certain adjustments made to the rules of the game. For example certain aspects of the game such as the imperfect information are removed with all aspects of the game presented in a face up manner creating a game of perfect information.

Roelofs (2012) also uses a Monte Carlo tree search to play the game. This paper can be seen to build upon the earlier work by Szita and further explores the possibility of playing the game using MCTS. Of note in this particular paper is the exploration of the different possibilities that present themselves for the node selection strategy and also

for the final node selections strategy. This paper also presents useful methods for breaking down the tree search into groupings in order to improve speed and help combat the issues that this author had with not being able to run enough simulations due to simulation speed. Results of the output from this paper were roughly equivalent to previous attempts to play this game with MCTS.

Another paper relating to the development of a bot capable of playing Catan is by Pfeiffer (2004) who develops a bot using reinforcement learning techniques. The method employed involves creating a set of high level strategies which can be picked from depending on the state of the game. These strategies are then used as heuristics to help guide the learning process for a self playing game which shows increased performance thanks to the learning process. The competency of the AI in this particular area is hard to judge as it is only really tested against a human player of which the skill level is unknown, though it would appear that this bot performs somewhat less effectively than the bots presented by Szita et al. (2009) or Roelofs (2012) but still achieves a reasonably acceptable level of play.

A further piece of work exploring AI for Catan is by Thomas (2003). The approach in this work is considerably different from the attempts which attempt MCTS as a method to play the game. Notably this work is based on a series of heuristics and high level strategies, drawing parallels to Pfeiffer's work but there is no reinforcement learning with a focus instead on low level heuristics to achieve the goals set out by a high level plan. The AI developed as a result of this paper is the basis of the AI in a large number of web servers that host the game as part of the JSettlers software. Its efficiency is noteworthy, with decisions of what move to play coming incredibly quickly.

More reading also shows a paper by Branca and Johansson (2007) This paper explores a multi agent heuristic driven approach to playing Catan. Notably this paper shows that their developments are not able to beat monolithic solutions with a single decision making unit, perhaps suggesting that this approach may be less effective than the others examined. In particular the AI struggled with the long planning that is required to control the game but did make good short term decisions. Of interest in this paper however is the ability of the bot to trade with other player, a feature that is commonly missing from other attempts to play Catan.

Other papers are less focussed on the develop of a bot that can play the game in a stand alone manner but are interested in aspects like determining optimal strategies for trading between players and analysing the most likely moves that a player could accept. For example a paper by Afantenos et al. (2012) analyses the conversations that occur between players and how this influences trading in the game.

More papers describing attempts to develop AI bots are difficult to find. Upon an analysis of the literature surrounding this game it appears that there are three main approaches taken to developing bots to play this game. Heuristic systems, MCTS based systems and reinforcement learning systems. It should be noted that most techniques, though mainly belonging to one of the above also commonly integrate features of the others and in fact it could be argued that MCTS is simply a form of reinforcement learning with the search training itself through the simulations of moves.

### 1.3 Aims

The aim of this project can be considered to build upon previous research into AI for this game and to design a bot that performs at a strong level. This could be classified by winning a high proportion of its games against bot opponents



## 2 Background

In order to allow better understanding of the game and the techniques in the project this section will explore useful background information relating to the game of Catan as well as various other aspects that will provide useful into the understanding of this project.

### 2.1 Catan

Catan is a popular boardgame designed by Klaus Teuber and first published in 1995. The game is considered a “Eurogame” a class of game often also described as “German Style”. The focus in this style of game is to put focus on player communication and skilful play. This could, perhaps, be contrasted with other board games in which more emphasis on conflict between players such as Risk or games where luck plays a much larger role in the proceedings such as Monopoly.

While comparing it with other games it is important to note that Eurogames such as Catan normally involves less skill based play than the more abstract and traditional games of go and chess. This is mainly owing to the absence of chance in these games as well as the general lack of communication and interaction between players when it comes to

In addition to the basic game that has been released there have been a number of optional expansions added to the base game which provide the opportunity for more players to join the game at once as well as allowing more complex and interesting scenarios to be explored. For the purposes of this project we will only consider the basic game played with 4 players and the most up to date version of the official rules.

Since its release the game has sold more than 22 million copies and been translated into at least 30 different languages (McNary 2015). Catan has also won numerous awards including the German “Spiel des Jahres” (Game of the Year) award and in 2005 was added to the Hall of Fame of the Games 100 buyers guide at the earliest opportunity (Spiel des Jahres) indicating its status as a critically acclaimed and popular title.

### 2.2 Rules of the Game

The rules of Catan are reasonably complex yet at the same time easy to understand. The following section will attempt to explain the rules of the game. If the reader already understands the rules then it is recommended that this section is skipped. An official source of the rules can be obtained from the Catan website. (<http://www.catan.com/service/game-rules>)

#### 2.2.1 The Board and Playing Equipment

The board of Catan consists of 37 hexagons, tessellated to form a larger hexagon where each edge consists of 4 pieces. The outer ring of hexagons on the board make up the tiles representing the sea. These tiles are considered out of bounds and nothing can be built on these tiles.

Each tile on the board that is not in the sea has a number and resource associated with it. These are the terrain hexes and there are 19 of them. The number on the tile corresponds to a result of double dice roll. No tile on the board has the number 7 on it. One tile in the board has no resource or number on it and is referred to as the desert. For each game the position of these tiles and the numbers on them is randomised, although some sort of agreement may be reached regarding the quality of a layout between players.

The board can therefore be thought to be made up of a number of hexes, nodes and edges. Hexes are the large hexagon shaped pieces which normally have a number and resource placed on them. Around each hexagon is 6 nodes on each vertex. These nodes are the positions at which settlements and cities in the game can be placed. Each of these nodes is adjacent to at most 3 and at least 1 labelled hex. Edges are the connections between two nodes and have roads built on them during the game.

There are 5 different types of resource in the game and are as followed. Each is required by the player to perform various actions in the game such as building. Alternative names for the resources are sometimes used and are provided for reference. In this report the first term listed will be used throughout.

- Clay (Brick) - 3 Hexes
- Wood (Lumber)- 4 Hexes
- Wheat (Grain) - 4 Hexes
- Sheep (Wool) - 4 Hexes

- Ore (Stone) - 3 Hexes

Around the board there are a number of ports. These ports are located in the sea hexes and there is one port for each resource as well as 4 generic ports to give a total of 9 ports. These ports are attached to either one or two territories and are used for certain cases of trades by the player.

Each of the 4 players in the game has an equal number of pieces that can be placed on the board. There is a fixed limit on the number of pieces of each type that a player can place onto the board determined by number of physical pieces that they possess.

- Road - 15 Pieces
- Settlement - 5 Pieces
- City - 4 Pieces

Not owned by the players are a deck of 25 development cards. These cards are shuffled before the game to randomise the order of them. Also not possessed by the player is a bank of resource cards which should be handed out as and when necessary by a nominated player.

### 2.2.2 Beginning the Game

At the start of the game one of the players is chosen to go first and play then proceeds in a clockwise direction. The start phase of the game of Catan is then begun. This phase has notable differences to the rest of the game. In turn each player places one settlement and one road on the board for free. The settlement may be placed on any legal node of the board. The road must be placed on one of the edges connecting to the node the settlement was placed on. After the 4th player has placed their road and settlement on the board then the second phase of the opening begins. The 4th player that has just placed their pieces then moves again and places another settlement and road. The settlement can again be placed in any legal location and the road must also be attached to the node the settlement was placed on. During this second phase however the player will also receive one of the corresponding type for each hex the settlement was adjacent to. This means that a maximum of 3 resources and a minimum of one resource can be obtained. This continues until the play has returned to the first player. A total of 8 settlements and 8 roads will have been placed over the board and regular play now commences from the first player and continues until the end of the game.

### 2.2.3 Regular Turn

During a regular turn a player begins by rolling the dice. The result of the dice roll will correspond to a value on the board with the exception of a 7. Resources are then distributed with each settlement adjacent to a hex with the rolled value on earning the owner 1 of the type of resource pictured on the tile. Cities earn the player two resources. It is possible for a player to own multiple structures around a single hex in which case they receive the number of resources corresponding to the structures around it.

After the dice has been rolled the player may spend any amount of time spending their resources to build or buy. They may also perform an unlimited amount of trades with other players or the bank. When a player is satisfied they may end their turn, resulting in play transferring to the next player in a clockwise direction. This type of turn continues until the game is won.

### 2.2.4 Building

On their turn a player may wish to spend their resources on building which allows them to place a piece onto the board. Any number of pieces may be placed in a single turn as long as the player has the piece to place and the correct number of resources. There are rules regarded the positions that each piece may be placed in:

- Road - 1 Clay, 1 Wood - Must be placed on an edge that does not already contain a road and is connected on at least one end to another of the players roads, settlements or cities. It may not go through an opponents city or settlement.
- Settlement - 1 Clay, 1 Wood, 1 Wheat, 1 Sheep - Must be placed on a road owned by the player. There must be a gap of at least 2 edges between the settlement being placed and any other settlement or city.

- City - 3 Ore, 2 Wheat - Must be placed on a settlement. The settlement piece is returned to the player after the city piece is placed.

Building may only take place during the player who is placing the pieces turn.

### 2.2.5 The Robber

The board has a robber piece on it that starts in the desert hex. After a 7 is rolled the robber must be moved to another hex by the player that has rolled the 7. If there are settlements or cities around the hex that the player has chosen to move the robber to then they may choose to steal a single resource from any of the owners of those structures. This resource is chosen randomly from the players hand. If there are no structures in the hex or the players that could be robbed do not currently own any resource cards then no resource shall be taken.

While a hex the robber located on it then it shall not produce any resources. Any time the number is rolled which corresponds to the value of the hex the robber is situated in then resources will not be given out to the owners of the surrounding settlements and cities.

If any player owns more than 7 cards when a 7 is rolled by any player they must give up half of their resource cards (rounded down) to the bank before any play continues. This rule is in place to stop players from hoarding resources.

### 2.2.6 Development Cards

As well as building a player may also spend their resources on development cards:

- Development Card - 1 Wheat, 1 Sheep, 1 Ore - If there are still cards remaining in the deck.

There are 5 different types of development card:

- Knight (Soldier) - 14 Cards - Move the robber and rob a resource
- Victory Point Card - 5 Cards - Add one victory point to your score
- Monopoly - 2 Cards - Nominate a resource - receive every card of the resource held by the rest of the players
- Road Building - 2 Cards - Place two road pieces for free
- Year of Plenty - 2 Cards - Receive any two resource cards from the bank.

A development card may be played at any time during a players turn, even before the dice has been rolled. Only one development card may be played on any given turn. This means, for example, that if a player plays a knight card before the dice is rolled that they may not play any other development cards they possess after they have rolled the dice. A player may buy any number of development card on their turn.

The victory point development card is an exception to the above restrictions on playing development cards. They may be played at any time, remain hidden until playing them would end the game and are not limited to having only one able to be played at a time.

### 2.2.7 Trading

When it is their turn a player may decide to initiate a trade. Trades may be made either with the other players or with the bank.

When trades take place with other players any combination of resources may be offered in return for any others being received. Development cards may not be traded. A player may only directly initiate a trade during their turn though it often transpires that in games between 4 human players in a live setting that this rule is often relaxed.

A player may also trade to the bank. When trading to the bank a specific number of the same type of resource are traded for a single one of the trading player's choice. The number of resources that must be traded to the bank to receive a resource is initially 4. This number can however be reduced by utilising the ports around the edge of the board. If a settlement or city is located next to one of these ports then they are said to belong to the player. If a player owns a generic port then the number of resources reduces to 3 for all types of resource. If a port displaying a specific resource is owned then the number required for that type of resource only is reduced to 2.

There is no obligation to accept a trade or ensure that trades are fair. However, all trades to the bank will be accepted as long as the correct amount of resources are given.

### 2.2.8 Winning the Game

A game of Catan is won when a player reaches 10 Victory Points. These points are obtained in a number of different ways. There are 3 ways that provide permanent victory points

- Settlements - Each settlement grants the owner 1 victory point
- Cities - Each city grants the owner 2 victory points
- Victory Point Cards - Each one owned by the player grants them 1 victory point. They remain hidden until they can be used to win the game or the game is over.

As well as these methods to gain permanent victory points there are also transient victory points which transfer between players as the game progresses.

- Longest Road - The player with the longest road of at least 5 pieces gains 2 victory points. Branches and loops are not counted in this total
- Largest Army - The player with the most knight cards played and has placed at least 3 gains 2 victory points

These victory points can be transferred to another player by them building a longer road or playing more knight cards than the player that currently has them.

## 2.3 Catan Strategy

In order to understand the development of a bot that can play Catan using AI techniques it is worth trying to understand the strategies and principles that players normally use when playing the game.

While there has been some work on trying to analyse Catan by applying Game Theory practices to it with notable work by Guhe and Lascarides (2014) in attempting to empirically analyse certain strategies and also by Keep (2010) who attempts to use mathematical theories to analyse gameplay and derive facts about the game. It is, however, clear that there is little work done on formally attempting to solve or analyse the game. This could be contrasted to games like checkers and connect four which have been solved.

### 2.3.1 General Principles

There are certain key principles that are generally followed during the game of Catan. Most obvious and important is the need to maximise the income of resources that a player receives throughout a game. Principally this is achieved in two different ways. Firstly this is done by making sure that settlements built are placed on nodes adjacent to hexes that are rolled more frequently. For example a 2 or 12 can only be produced in one way by two dice while an 8 or 6 can be rolled 5 different ways. A table is provided below of the probabilities of these rolls taking place. Secondly, resource gain is increased by building more settlements and cities. It is generally taken that they should be built whenever possible in order to increase the flow of resources for a player. A minor consideration that should also be taken into account is the spread of different values covered; if a player can place their settlements so that they cover a wide variety of values then they will make their flow of resources more consistent, allowing for fewer missed opportunities in turns.

Another crucial aspect regarding resources is to ensure that a player has a good spread of different types of resources. If a player is too focussed on few resources types then they will not be able to progress at the same rate that another player is as they will need to trade in order to make up for the shortage in resources.

Also important is that the resources a player possesses are constantly being spent. This is important for two main reasons. If a player hoards resources then they run the risk of having to discard half of them if the robber is activated, wasting them. Additionally, if a player holds resources rather than spending them they are wasting the opportunity to gain from those resources.

A further key principle that should be observed is the synergy of certain types of resources with one another. Everything that can be purchased with wood also involves clay to be built and everything that can be purchased with ore also requires wheat. This means that it can be very effective to make sure that we try and keep these resources in pairs as they tend to be considerably less effective without their complimentary resource.

The relative value of different resources also changes throughout the course of the game. Clay and wood are generally considered to be early game resources as they are vital for the building of roads and settlements which are generally placed at the start of a game. Wheat and Ore are necessary for building cities and buying development cards which tend to be more useful as a game approaches its conclusion. All of these factors must be taken into account when deciding on the location and choice of resource deployment.

### 2.3.2 Common Human Strategies

When playing the games that are certain strategies that are commonly followed and recommend to new human players of the game. These strategies are generally the ones that are implemented when bots are designed that attempt to capture the tactics and strategies adopted by expert human players.

There are, generally speaking, 2 main strategies that are employed by most players in the game. The first of these is the wood and clay strategy. This strategy normally ensues if a player has a strong income source of wood and clay. When following this strategy focussed is placed on rapidly expanding the number of settlements that the player owns as well their roads. A main objective of this particular strategy is to obtain the victory points for owning the longest road. The large amount of roads that can be placed also allow the opportunity to control large sections of the board and block opponents off. This strategy generally becomes weaker later in the game where the maximum number of roads and settlements have been placed, rendering the initial investment in wood and clay eventually ineffectual. It is important that the expansion settlements built fill the voids in resource production left by focussing on clay and wood.

In contrast to the wood and clay strategy is the ore driven strategy. This particular method involves focussing primarily on ore and also on wheat. The motivation behind this technique is to upgrade settlements to cities and also to purchase development cards. While the brick and clay strategy focusses on getting the longest road this methods eschews that and instead focusses on getting victory points from development cards, both through the victory point cards directly and also from the largest army victory points which results from playing the knight cards the player obtains. This strategy tends to be more effective during the late stage of the game, where the opportunities to build new roads and settlements would be limited but there are plenty of development cards remaining.

Naturally a hybrid strategy also exists, focussing on a mixture between these two strategies. Generally this involves getting a good mix of all of the resources and trying to balance building roads and settlements with upgrading those settlements to cities and buying development cards. This strategy is more opportunistic than the other two with less emphasis placed on obtaining the largest army of longest road victory points that correspond to the other two strategies.

It is important to note that these strategies must be flexibly chosen and that it is not possible to decide on a particular course of action before the layout of the board is seen and the order of play determined. Only after this information and the knowledge of other players actions is taken into account is it possible to determine a strategy that will be played, this presents difficulties from an AI perspective as it will not be possible to hard code a single effective strategy into a bot which it can follow.

### 2.3.3 Opening Turn

Particularly crucial in Catan is the opening moves played each player at the start of the game. These opening moves result in two settlements and two roads being placed, for free, for each player and decide the strategy that will adopted by the player for a considerable portion of the game.

As these settlements can be placed in any legal location on the board without the requirement for roads it is important to ensure that a good position for the settlements is chosen. Ideally this should be occupying as many commonly rolled values as possible and should also be on resources that work well together.

Unlike other aspects of the game the strategy that should be employed in this section of the game is possible to capture with a set of rules and can be described in a mechanical fashion. This could be helpful considering the differences in this stage of the game compared to other parts.

### 2.3.4 Robber Strategy

A further subsection of the strategy of the game that must be considered is the placement and usage of the robber piece.

One part of this aspect of the game is deciding on where to position the robber when the player rolls a 7 or plays a knight card. Clearly a player will not wish to deprive themselves of resources, except in very rare and specific situations. Normally the piece is placed so that the maximum amount of disruption to a players resource collection is caused. This is worked out by taking into consideration the likelihood of the value on that particular hex being rolled as well as the number of structures around the nodes of that hex.

When moving the robber to a different hex a further consideration must be made as to whom the ideal player to deny resources is. This is often the person closest to winning the game, especially if the game is near its conclusion.

If no player is close to winning the game then it is usually a good decision to try and deny the most resources possible.

A more minor choice that must be considered is the decision of what player to rob in certain circumstances. Normally this decision is neither particularly meaningful nor hard. If a player is likely to have a resource that is needed then that player is usually chosen otherwise a random choice can be made.

### 2.3.5 Trading

As well as dealing with the sub-strategy regarding the placement of robbers throughout the game the player must also formulate strategies that can be followed with regard to trading.

One type of trading that occurs during the game is player to player. Trades of this type are generally acceptable if there is some benefit to both sides during the trade. It is possible that imbalanced trades may be completed. These trades provide some interesting possibilities. Firstly they allow a trade to be given increased leverage to someone in return for a sorely needed resource. Another part of imbalanced trades is that they can be used to discard resources for some return and reduce the risk that the player will have to discard resources if a 7 is rolled and they possess too many resources.

A degree of research has been done into player to player trading in Catan. The paper by "CITATIONS HERE" ....

The other type of trading that happens in Catan is bank trading. In this type of trading 2-4 resources of a single type are given to the bank in return for a single resource of any type. There are both positives and drawbacks to this type of trade. On one hand this type of trade allows the player to obtain resources that are not currently held by any player in the game and also allows them to get these resources without benefiting any other player. Negatively however, this type of trade, especially if undertaken at a 4:1 ratio rarely represents the most effective use of the resources being traded. A good trade strategy must balance when bank trades are made and when trades to other players are made.

## 2.4 Common AI Techniques for Game Playing

There are a number of common techniques that are often employed when creating game playing bots. It is worth considering these when it comes to designing a bot that can play Catan.

### 2.4.1 Minimax Search

The minimax algorithm is a tree search that uses brute force and state evaluation in order to choose a move. Minimax gets its name from the attempt to minimise the loss that can occur in the worst cases. The algorithm takes the form of a search tree with every node in the tree representing a possible state of the game. Normally the search is used for a two player game with each layer, or ply, of the tree equivalent to the states that could be present in a player's turn. The edges in the game represent the moves a player could apply to return to a state.

A key part of minimax is the evaluation function applied to all leaf nodes. Its role is to assign a value to all of the nodes depending on the state they are in. An evaluation function could be as simple as recording whether the game is won or lost but likewise could involve a complex evaluation of the state of the game in order to provide a more in depth analysis of the position.

If the opponent in a game is playing rationally then minimax has been proven to result in optimal play if the full search space of the tree is explored and the evaluation function is admissible.

Minimax has been adapted to work for games that have non-determinism present in them. There are several different approaches to this scenario by they are commonly handled by incorporating nodes that have probabilities attached to them reflecting the results of the random event such a dice roll. These nodes are normally referred to as chance nodes.

The most notable downside of minimax search is that it is quite inefficient, even when nodes that can be ignored in the search are removed from the tree using techniques such as Alpha-Beta pruning. Generally any games that have a reasonably high branching factor, the average number of moves that can be played on a single turn, are infeasible to be solved by a minimax search. Chess for example has an average branching factor of about 35 while for go this number is 250. The time complexity of minimax is essentially the same as a depth or breadth first search, namely  $O(b^d)$  where  $b$  is the branching factor of the tree and  $d$  is the depth of the tree. This effectively means that a game with a large number of turns before completion will not be able to have its state space fully explored. For minimax to be used in a game like this a depth limit is normally set, which in turn reduces the search to be less than optimal. On games such as go with an extremely high branching factor this leads to a very shallow tree which in turn gives poor performance.

### **2.4.2 Monte Carlo Tree Search**

Monte Carlo Tree Search (MCTS) is a technique that builds on the tree exploration approach of minimax. A key benefit of MCTS is that it can be applied to games with a large branching factor where the performance of minimax would be poor and produce good results.

MCTS builds upon the Monte Carlo method and applies it to trees. The Monte Carlo method relies on repeated random sampling in order to produce a result and is commonly used in scenarios where an exact answers would be infeasible or impossible to find.

### **2.4.3 Expert Systems**

### **2.4.4 Reinforcement Learning**

### **2.4.5 Evolutionary Algorithms**

## **2.5 JSettlers2**

To aid in the development of this project a 3rd party piece of software, JSettlers2 was used in order to run the engine of the game and evaluate the performance of any bots developed. This software is open source and freely available to download and modify.

There are several benefits to using this particular software as the basis for developing the AI on. Firstly it can be observed that several other papers have used this API in order to develop their projects. This means that it would theoretically be possible for the bot developed in this project to be able to play the developments of other authors in order to test their effectiveness.

As well as being able to have the ability to play against other bots that have been produced on the same platform it will also be possible to use the bots that are built into JSettlers. These bots will be useful for measuring the performance of our bot. The development of these bots was mainly undertaken by PERSON HERE as part of their thesis and play in a rule driven heuristic manner.

Another key benefit of using the JSettlers2 software is that we will not have to program the logic of the game ourselves, this will save time and allow more time to be spent on developing the AI. A GUI is also included in the game engine, allowing the visualisation of games and the ability for humans to play. This will permit qualitative evaluation of bots to take place.

There are several other properties of JSettlers2 that make it an effective tool for usage during this project. One of these is the abilities to run a set of automated games without the need for intervention by the user. The results of these automatic games can then be recorded allowing for evaluation of the bot.

### **2.5.1 JSettlers2 Structure**

JSettlers2 follows a server client architecture with communication between the two types of system taking place using sockets. The server is hosted on a single computer and is responsible for maintaining the state of the game and communicating with the clients. The clients do not necessarily need to continually maintain a state of the game and only need to send messages in order to reflect their decisions to the server.

Communication takes place in the form of messages following a protocol specified in the documentation of the software. These messages cover all aspects of the game and are used to send updates and choices to the server and also to receive the actions of other players.

In game representation of the board and other objects in JSettlers takes a relatively simple approach. A board is represented using a system of hexadecimal coordinates, a board contains a list of the structures placed on it along with details of the owner and location. From this there is enough information for the engine to calculate the validity of a player's moves and the eventual winner of a game. Further objects exist in the game to represent a player and the resources and development cards that they possess.

## 3 Implementation and Development

### 3.1 Overview and Plan

### 3.2 Interaction with API

### 3.3 Opening move generator

### 3.4 Move generator

### 3.5 Monte Carlo Tree Search

#### 3.5.1 Handling Non-Determinism

#### 3.5.2 UCT Formula

#### 3.5.3 Application of expert knowledge

### 3.6 Simulator

#### 3.6.1 Simulator Heuristics

#### 3.6.2 Simulator Fidelity

#### 3.6.3 Simulator Performance

### 3.7 Optimisation



## 4 Results

### 4.1 Effects of Heuristic Opening Strategy

### 4.2 Effects of Adjustment of Simulation Count

### 4.3 Effects of Adjustment of UTC Exploration Parameter

### 4.4 Play against Humans

### 4.5 Qualitative analysis of Play

## 5 Evaluation

### 5.1 Performance

### 5.2 Comparison with Other Systems

### 5.3 Limitations and Issues

## 6 Conclusion

### 6.1 Possible Future Work

## References

- Afantenos, Stergos et al. (2012). “Developing a corpus of strategic conversation in the settlers of catan”. In: *SeineDial 2012-The 16th Workshop On The Semantics and Pragmatics Of Dialogue*.
- Branca, Luca and Stefan J Johansson (2007). “Using multi-agent system technologies in settlers of catan bots”. In: *City 2*, p. 3.
- Guhe, Markus and Alex Lascarides (2014). “Game strategies for the Settlers of Catan”. In: *Computational Intelligence and Games (CIG), 2014 IEEE Conference on*. IEEE, pp. 1–8.
- Keep, Peter (2010). “Settlers of Catan Analysis”.
- McNary, Dave (2015). *Settlers of Catan Movie, TV Project in the Works*. URL: <http://variety.com/2015/film/news/settlers-of-catan-movie-tv-project-gail-katz-1201437121/> (visited on 03/31/2017).
- Pfeiffer, Michael (2004). “Reinforcement learning of strategies for Settlers of Catan”. In: *Proceedings of the International Conference on Computer Games: Artificial Intelligence, Design and Education*.
- Roelofs, G (2012). “Monte carlo tree search in a modern board game framework”. In: *Research paper available at umimaas.nl*.
- Speil des Jahres. “Die Siedler von Catan”. URL: <http://spieldesjahres.com/en/die-siedler-von-catan> (visited on 03/31/2017).
- Szita, István, Guillaume Chaslot, and Pieter Spronck (2009). “Monte-carlo tree search in settlers of catan”. In: *Advances in Computer Games*. Springer, pp. 21–32.
- Thomas, Robert Shaun (2003). “Real-time decision making for adversarial environments using a plan-based heuristic”. In: