Universiteit Leiden
The Netherlands

Opleiding Informatica

Comparing a realistic player for the game The Great Dalmuti against a model player with superhuman capabilities

Thomas Rooskens (s3339661)

Supervisors:
Max van Duijn & Ramira van der Meulen

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl 4/7/2024

Abstract

Humans entertain themselves in several ways, one of which is playing card games, it can however be hard to find optimal strategies for complex card games. In this thesis we model a superhuman an a realistic player for the card game the Great Dalmuti to discover and confirm different strategies. We will take a look at the capabilities of the players and how they are implemented. We will also take a look at how the different players make use of different strategies. In the results section we will see how each of the players performs against a player that always plays their worst set of cards. The most interesting result that can be derived from these results is that the set chaining strategy has a large positive impact on the position distribution of a player. This means that trying to make use of the set chaining strategy in your own games could help you place first more often. Future work could be dedicated to creating a deep-reinforcement learning agent / algorithm that also decides when it is a good play to pass.

Contents

1 Introduction

Card games are one of the many ways with which humans entertain themselves. These games are mostly played with a group of friends or relatives and often contain a competitive element. If humans want to win these competitive games, they have to come up with a strategy that enables them to outcompete their competitors. However, finding the best strategy is hard. Many games contain elements that benefit from having a good memory, or from calculating the probability of events within the game. Human memory is limited [Cow10], and calculating the exact probabilities on the spot is hard. We have seen in the past that computers have beaten humans in many board games such as AlphaGo, which has beaten a world champion at the board game Go [WZZ+16]. In this thesis we will take a look at the card game the Great Dalmuti and try to uncover or confirm strategies that work using a player with superhuman capabilities and a player that plays human-like.

The Great Dalmuti is a game that can be played with 4 to 8 players. The goal of the game is to be the first player to play every card in hand, so that one becomes the Greater Dalmuti. To play a set of cards, two conditions must be met. First, the set of cards on the table and that are being played are of equal size. Second, the rank of the set of cards being played is lower than the set of cards currently on the table. Playing three 3's on top of three 4's is therefore a valid play, while the other way around is not.

The game the Great Dalmuti might seem like an easy game in hindsight, but deciding whether and which set of cards you should play depends on a few factors. For instance, making a decision depends on which cards we have, which cards our opponents have, which cards have been played, how many players have passed already, etc. The Great Dalmuti is a great card game to use statistics to make decisions, because there are so many factors to take in to consideration when making a decision.

The source code for this research can be found at: `https://github.com/Thomsr/dalmuti`.

1.1 Research Question

The main objective of this bachelor thesis is to uncover strategies that players can use to increase their chance of becoming first. To achieve this, we will first create a player with superhuman capabilities so that there is a player that is very good at the game to compare our realistic player model against. The more often the realistic player achieves a higher ranking than the player with superhuman capabilities, the better the strategy is that it is using.
The objectives of this thesis can be divided to the following sub-research questions:

1. How do we create a player with superhuman capabilities?
2. How do we model a realistic player?

The main research question of this thesis is: 'Comparing a realistic player for the game The Great Dalmuti against a model player with superhuman capabilities.'

We will answer the research question and the sub-research questions by first providing some background on partially observable card game strategies and how to model a realistic player. This will be followed by a methods overview in which we will explain how to play the Great Dalmuti, how the superhuman player works and how the realistic player works. Afterwards we will take a look at the impact different strategies have on the players and how the superhuman player compares to the realistic player. Afterward we will interpret the results in the discussion and conclusion section. Lastly we will talk about potential future work that can be done.

2 Background Information

2.1 Partially observable card game strategies

There are a lot of card games, many of which are partially observable such as: poker, blackjack, and UNO. A card game being partially observable means that a player has limited access to information about the current state of the game. For instance, when someone plays black jack, they do not know which card will be drawn next. Determining the best strategy for each state in a partially observable game is significantly more challenging than doing so in a fully observable game. This is because when we use a strategy for a fully observable game, we know the outcome of our strategy beforehand. This is not the case in a partially observable game, here we do not know the result of our action.

Most card games also contain a competitive element by letting players play against each other. The card game is now also stochastic with the introduction of other players since we do not know the probability of their actions. There has however been an attempt to find a best-response in a partially observable card game. In this thesis the authors try to find a best-response in a partially observable simplified poker game. Here they tackle the problem by modeling it as a partially observable Markov Decision Process (POMDP) [OSV]. This research is applicable on the Great Dalmuti since it is also a partially observable stochastic card game. For instance, in the game the Great Dalmuti, we do not know which cards our opponents have and which set of cards they will play next. This makes the Great Dalmuti a good card game to use a POMDP on.

2.2 Modeling a Realistic player

2.2.1 Q-learning

Creating a realistic player that that performs like a human can be quite challenging. This is because it is often challenging to predict the actions of humans. People have however been comparing humans against machines [ICDEC+11]. In this paper the authors compare humans against Q-learning [WD92] in 20 tests with each containing 7 exercises. The results show that there is no significant difference in how the humans and Q-learning perform in the 7 exercises. This means that Q-learning or other reinforcement learning algorithms might be a good way to model a realistic player for the game the Great Dalmuti since it is able to perform the same as a human would.

2.2.2 Working memory

While playing card games, humans use their working memory to store important facts about the current game state. For instance, in the game the Great Dalmuti, a player might count how many 2's and 3's there have been played. Knowing when all these cards have been played can give a competitive edge against your opponents. The working memory is however not infinite, meaning that a human player is unable to memorize every card that has been played. On average a human player is able to fill 3-5 'slots' of working memory [Cow10]. This does not mean that a human player is able to memorize 3-5 'numbers'. They can store the information in a more conceptual way, for example, human players can memorize the number of times 3-5 card types have been played as a 'chunk' of information [CCM16].

3 Methods

3.1 The Great Dalmuti

As stated in the Introduction[1] the game we have used to compare our agents is the Great Dalmuti. The goal of the game is to shed your cards as quickly as possible, while obeying the conditions of the game and accounting for the plays of your opponents.

The Great Dalmuti has a deck of 80 cards, each with their own rank. The cards are distributed as follows:

Card	Rank	# of cards in deck
Jester (wildcard)	13	2
Peasant	12	12
Stonecutter	11	11
Shepherdess	10	10
Cook	9	9
Mason	8	8
Seamstress	7	7
Knight	6	6
Abbess	5	5
Baroness	4	4
Earl Marshal	3	3
Archbishop	2	2
Dalmuti	1	1

3.1.1 How to play

Setup To determine the order of play, all players draw 1 card. The players are then ranked from best card to worst card, the worst card being the card with the highest rank. Players who have drawn a card with the same rank draw another card.

Some players get a title based on their placement. The player in first place is called the Greater Dalmuti, the player in second place the Lesser Dalmuti, the player in second last place the Lesser Peon and the player in last place the Greater Peon.

Starting a game The Greater Peon shuffles and deals the cards starting with the Greater Dalmuti until the deck is exhausted. This means that it is possible that some players have more cards than others. For example, when playing with 5 player each player gets 16 card, but when you play with 6 players, the first two players get 14 cards and the other 4 players receive 13 cards [1]. It is difficult to determine if it is better or worse to receive more cards, it depends on which card is the 'extra' card. Before each game after the first one, a card exchange takes place. The Greater Dalmuti gives 2 cards of choice to the Greater Peon in exchange for the 2 best cards of the Greater Peon. The

[1]It is possible that both jesters are given to the Greater Peon. In this rare case, the roles of the positions become reversed. The Greater Peon becomes the Greater Dalmuti, the Lesser Peon becomes the Lesser Dalmuti, etc. This mechanic is ignored in our research due to its negligible impact on the rest of the game.

same happens between the Lesser Dalmuti and the Lesser Peon, except they exchange 1 card. The Greater Dalmuti starts the game by playing any set of cards.

Playing a round A player has a few options when it is their turn:

- They can play a set of cards that have a higher rank than the cards currently on the table;
- They can play any set of cards when there are no cards on the table;
- They can pass their turn to the next player at the table;

A round ends when, every player except for the player who last played a set of cards, passes. A new round is started.

Ending a game A game ends when every player except for 1 has shed their cards. Players then get seated in the order that they finished and a new game begins.

3.2 Player with superhuman capabilities

The player with superhuman capabilities (a_{super}) is defined as follows:
The player has:

- Enough memory to memorize every card played in a round.
- Enough computing power to compute the probabilities necessary to make a decision.
- A few strategies and assumptions such as set chaining (section 3.2.3), playing a set of cards that you can also close (section 3.2.4) and assuming a player never splits their cards (section 3.2.5).
- A deterministic play style.
- The same access to information as the other players on the table. This means that the player knows which cards have been played and how many cards each player is holding. The player does not know which cards the opponents are holding.

Since a_{super} is able to memorize every card that has been played, we can calculate the chance that a player has a certain set of cards in their hand. To calculate this, we use the formula for a hypergeometric distribution:

$$P(X=k) = f(k;N,K,n) = \frac{\binom{K}{k}\binom{N-K}{n-k}}{\binom{N}{n}} = \frac{\frac{K!}{(k!(K-k)!)} * \frac{(N-K)!}{((n-k)!((N-K)-(n-k))!)}}{\frac{N!}{(n!(N-n)!)}} \tag{1}$$

where:

k = number of successes in sample
N = population size
K = number of successes in population
n = sample size

Suppose we want to calculate the probability that 3 out of the 9 cards the player $a_{opponent}$ is holding are a 5. Additionally, suppose that 14 cards have already been played and that one of them was a 5. This means that there a total $80-14=66$ cards left, with four of them being a 5.

To calculate the probability, we fill in the hypergeometric distribution formula:

$$P(X=3)=f(3;66,4,9)=\frac{\binom{4}{3}\binom{66-4}{9-3}}{\binom{66}{9}}=\frac{\binom{4}{3}\binom{62}{6}}{\binom{66}{9}}\approx 0.00664$$

This means that there is a 0,664% chance that $a_{opponent}$ has three 5's in their hand.

A set of cards is considered as a good set of cards when: you can easily get rid of the set of cards and when it is hard to for other players to play on top of the set of cards. Knowing this, we can calculate two values using the hypergeometric distribution formula:

3.2.1 Play Chance

a_{super} can calculate the probability that a set of cards can be played. The player does this by calculating the probability for each player to have a set of cards with a higher value than the set of cards we want to play. The probability that a player has a set of cards of a higher value is then divided by the places the player sits away from the a_{super}. Do this for each player currently playing, and you get the probability that a set of cards can be played. The formula for getting the play chance is as follows:

$$P_{play}(c,a,T)=\sum_{c'=c+1}^{12}\left(\sum_{n=1}^{N-1}\frac{f(a;T,c'_{left},n_h)}{n}\right) \tag{2}$$

where:

c = the card we want to calculate the play chance for
a = the amount of cards
T = the total number of cards left
N = the number of players in the round
c'_{left} = the number of card c left
n_h = the number of cards in player n's hand

Suppose we have a set of three 9's and want to calculate the play chance of this set. There are 4 players (including us) in the current round and and there are a total of 44 cards left.
We observe the following relevant information (note that 9 itself is not included because we can't play a 9 on top of a 9):

Card	Amount left
12	7
11	5
10	4

Player	Number of cards in hand
1	7
2	9
3	5

We can use this information to calculate the play chance of three 9's:
$P_{play}(9,3) = (0.00954 + 0.02166/2 + 0.00287/3) + (0.02146 + 0.04602/2 + 0.00682/3) + (0.06032 + 0.11477/2 + 0.02146/3) \approx 0.192939$

3.2.2 Round Close Chance

Similary a_{super} can calculate the probability that a set of cards will close the current round. This is done by calculating the probability that other players have a set of cards in their hand that is of a lower rank than the set of cards we want to play.

$$P_{roundClose}(c,a,T) = \sum_{c'=1}^{c-1} \left(\sum_{n=1}^{N-1} f(a; T, c'_{left}, n_h) \right) \tag{3}$$

where:

c = the card we want to calculate the play chance for
a = the amount of card c
T = the total number of cards left
N = the number of players in the round
c'_{left} = the number of card c left
n_h = the number of cards in player n's hand

Suppose we want to also calculate the round close chance of three 9's. We observe the following relevant information:

Card	Amount left
8	6
7	5
6	4
5	3
4	2
3	2
2	2
1	0

Player	Number of cards in hand
1	7
2	9
3	5

We can use this information to calculate the round close chance of three 9's:
$P_{roundClose}(9,3) = \sum_{c'=1}^{9-1} \left(\sum_{n=1}^{4-1} f(3; 44, c'_{left}, n_h) \right) \approx 0.247473$

We can use these two chances to determine how good a set of cards is. A set of cards is a good set if the Play Chance is high, and the Round Close chance is low (a value closer to 0 is better). The a_{super} calculates how good the possible playable sets of cards are and plays the set of cards that have the worst value.

a_{super} also uses some simple strategies to determine which set of cards it should play next.

3.2.3 Set Chaining

a_{super} always checks if all but one set of cards in their hand have a round close chance of 0 (meaning that a_{super} is assured that no one can play a set of cards with a higher rank). If this check is true, then that means that the player is able to get rid of their hand if they play the set with a $P_{roundClose} \geq 0$ last.

Suppose we are the first to play in a round and have the following cards in our hand with their corresponding $P_{roundClose}$ values.

Set	$P_{roundClose}$
3x 9	1.2
2x 5	0
1x 1	0

Since we won the previous round, we can choose any set of cards we want to play. Normally we would play 3x 9 because it has a $P_{roundClose} > 0$. However, this is the worst set of cards we could play since we are able to get rid of all our cards. The right order should be to either play 2x 5 or 1x 1 because we know that these sets of cards win us the round. After we have played both the 2x 5 and 1x 1 we can play the 3x 9, thus getting rid of all our cards.

3.2.4 Playing a set of cards that you can also close

Another strategy that the a_{super} uses is when a_{super} needs to start a round. It tries to play a 'bad' set of cards which it knows it can close with another set of 'good' cards. This results in the player being in control of the game and other players possibly 'wasting' their cards.

Suppose we have the following sets of cards in our hand with their corresponding $P_{roundClose}$ values.

Set	$P_{roundClose}$
2x 12	4.8
3x 11	3.2
2x 9	2.9
4x 5	0.8
3x 4	0

In this situation we would normally play the 2x 12 because they have the worst $P_{roundClose}$ value. However, it would be better to play the 3x 11 because we know that we can also close the round with our 3x 4. This keeps us in control of the game and let's us play the 2x 12 after we have won the round.

3.2.5 Assume a player never splits their cards

An assumption a_{super} makes is that the opponents never split their cards. This is because splitting cards is almost always a bad decision. We will see proof of this statement in the results sections 4.1.4. Because of this assumption, we need a new formula to calculate the probability that a player has a set number of cards in their hand:

$$P(a, T, c_{left}, n_h, n_{cPlayed})_{set} = \begin{cases} 0, & \text{if } n_{cPlayed} > 0 \\ f(a; T, c_{left}, n_h), & \text{otherwise [1]} \end{cases} \quad (4)$$

where:

a = number of successes
T = total number of cards left
c_{left} = number of card c left
n_h = number of cards left in player n's hand
$n_{cPlayed}$ = number of card c's player n has played

3.3 Realistic player

The realistic player ($a_{realistic}$) is defined as follows: The player has:

- limited memory, about 3-5 working memory slots in which the player can store the number of times a certain card has been played.
- Not enough computing power to calculate the exact probabilities necessary to make a decision.
- The same strategies as a_{super} 3.2, such as set chaining, playing a set of card that you can also close and assuming a player never splits their cards.
- A deterministic play style.
- The same access to information as the other players on the table. This means that the player knows which cards have been played and how many cards each player is holding. The player does not know which cards the opponents are holding.

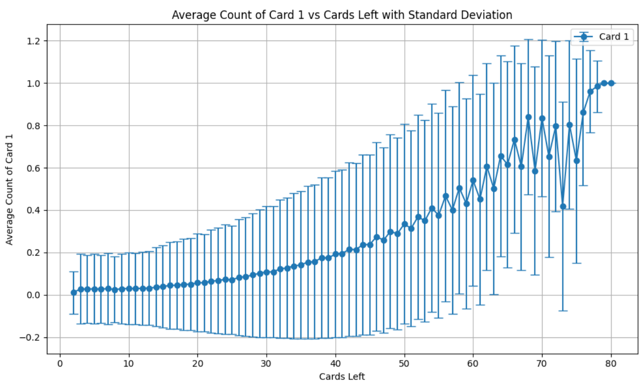
3.3.1 Memorizing cards

It is impossible for $a_{realistic}$ to memorize every card that has been played since $a_{realistic}$ has about 3-5 working memory slots available to store this information. We therefore need to decide which cards we should memorize. The best cards to memorize are 1, 2 and 3. These cards are considered the 'best' because they can be played on a wide range of other cards, while few cards can be played on top of them. We will therefore use the first 3 working memory slots to store how many 1's, 2's and 3's are left in the game

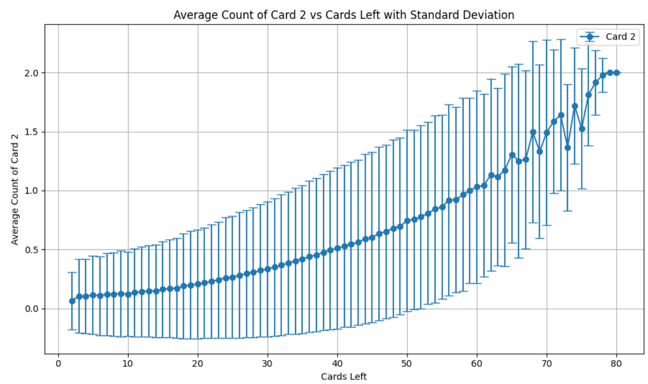
3.3.2 Using heuristics to approximate card values

We can see in section 4.1.1 that being able to calculate the exact probabilities is great for calculating when we can use certain strategies. However, $a_{realistic}$ does not have enough computing power to calculate these probabilities. This means that the realistic player will have to make use of heuristics to approximate the value of a set of cards.

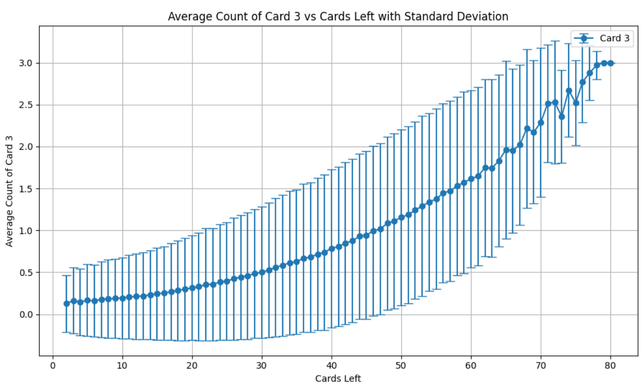
To create these heuristics, we will take a look at how many of each card is left versus the total number of cards left across 50.000 games played with 4 a_{super} players.



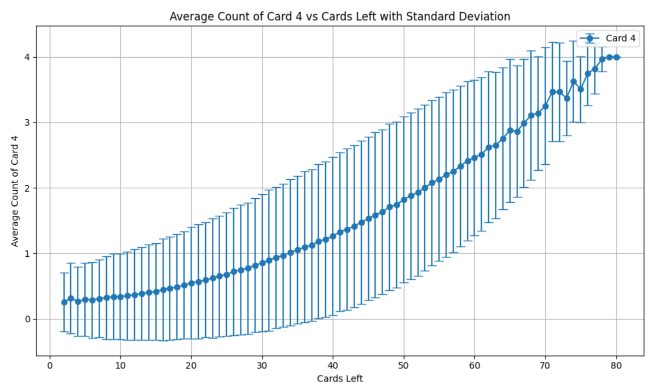
(a) Card 1



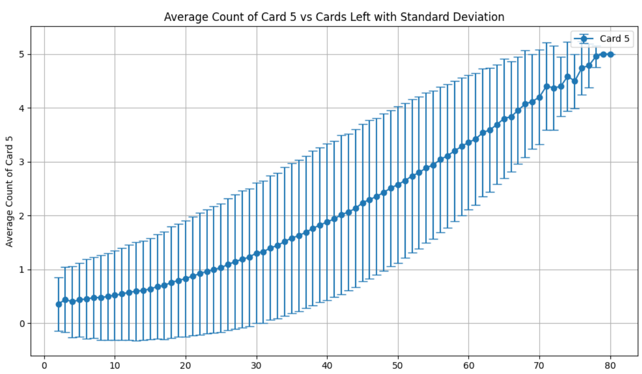
(b) Card 2



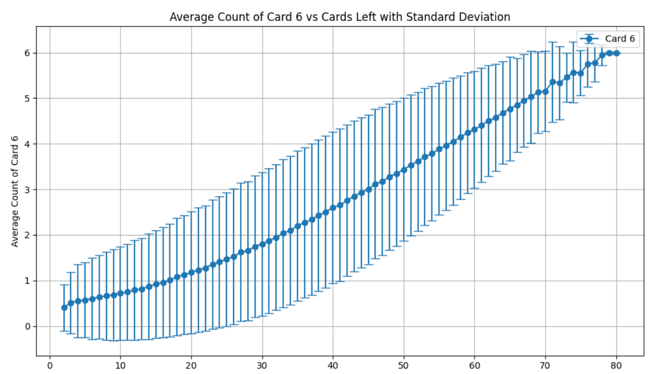
(c) Card 3



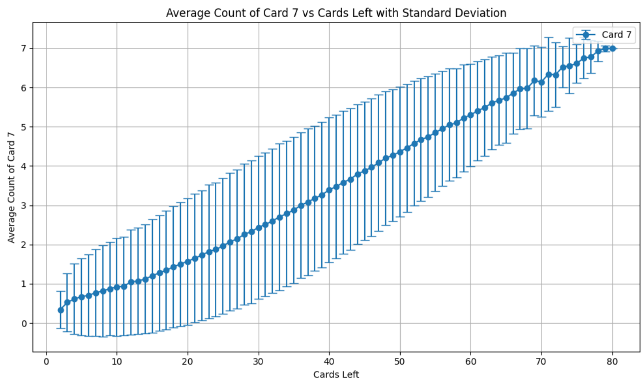
(d) Card 4



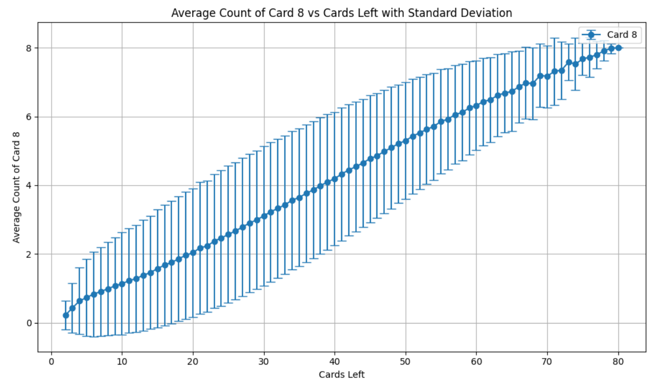
(e) Card 6



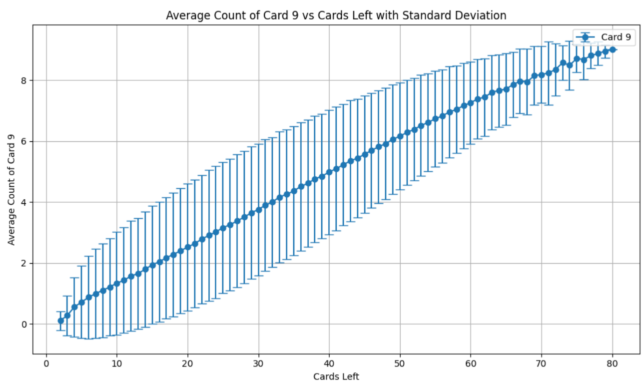
(f) Card 6



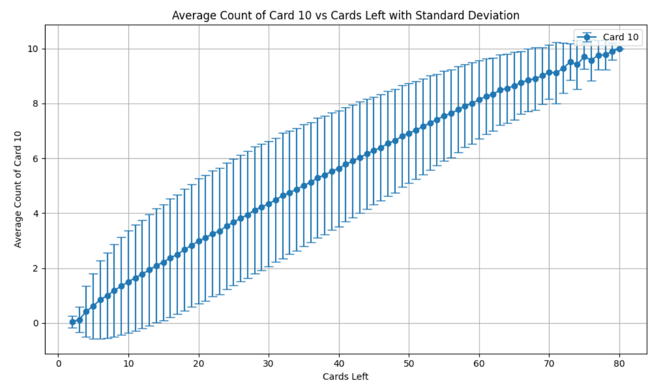
(g) Card 7



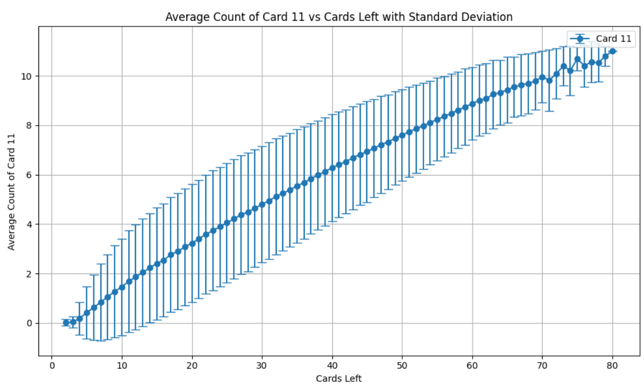
(h) Card 8



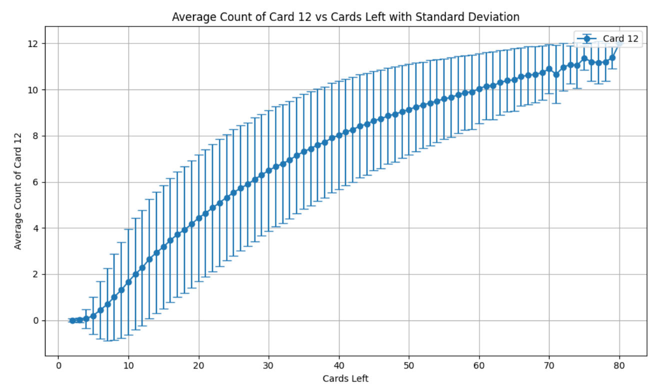
(i) Card 9



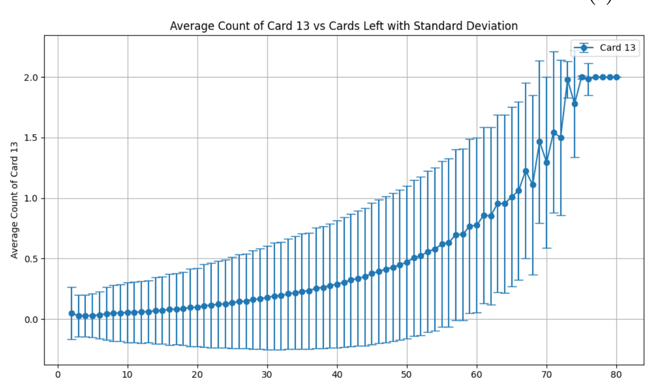
(j) Card 10



(k) Card 11



(l) Card 12



(m) Jesters

Figure 1: These graphs show for each card type, how many of that card type's cards are left on average in comparison with the total number of cards left. The results are obtained from 4 a_{super} players playing 50.000 games.

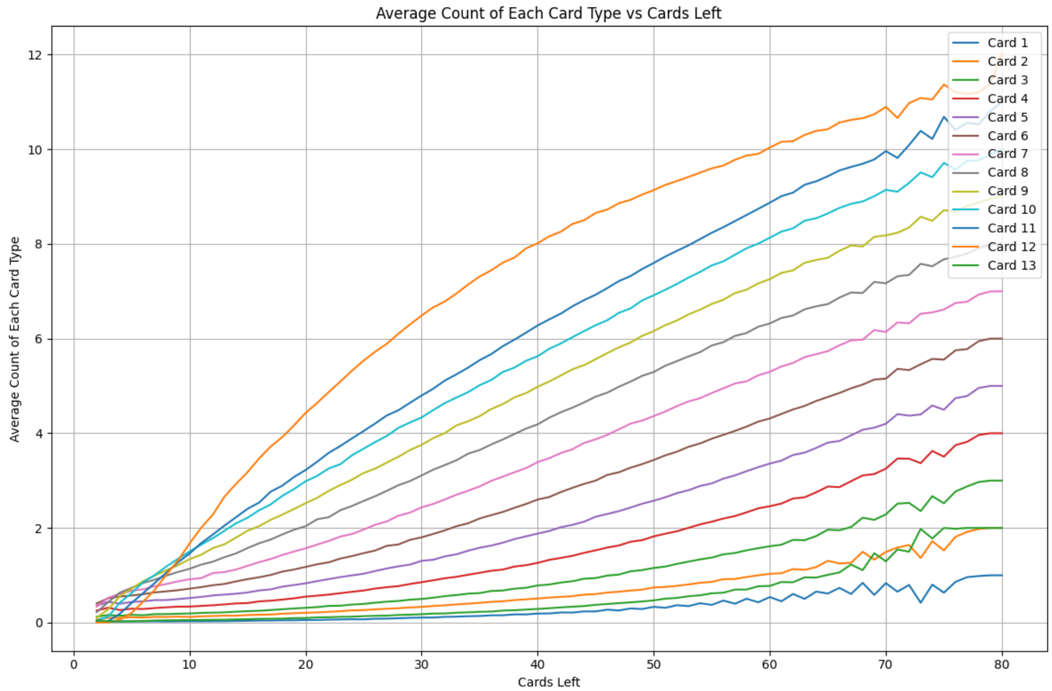


Figure 2: This graph shows for each card type, how many of that card type's cards are left on average in comparison with the total number of cards left. The results are obtained from 4 a_{super} players playing 50.000 games.

Using figure 2 we can read how many cards of each card type are left in comparison with the total number of cards left. For instance, if we want to know how many 9's are left at 40 total cards left, we can see from the graph that approximately 5 of the 40 total cards left are a 9.

Figure 1 shows a more in depth graph for each of the card types, it shows the standard deviation for each point on the graph. For instance, figure 1g shows for card type 7, that there are between 1.7 and 5.1 7's left when there are a total of 40 cards left.

We can use the information from these graphs to create heuristics for the realistic player.

3.3.3 Simple heuristics

A simple set of heuristics that can be used to approximate how many of each card are left are as follows: At approximately 80, 40 and 20 total cards left:

Card type	cards left, total ≤ 80	cards left, total ≤ 40	cards left, total ≤ 20
4	4	2	0
5	5	2	1
6	6	3	1
7	7	4	2
8	8	4	2
9	9	5	2
10	10	6	3
11	11	6	4
12	12	8	4
jesters	2	1	0

Note that there are no heuristics for the ranks 1, 2 and 3. This is because $a_{realistic}$ is able to memorize how many of these cards are left 3.3.1.

3.3.4 Approximation functions

Another way to estimate the number of cards remaining is to use a function that takes the total number of cards left as input and outputs the number of cards of a specific type remaining. In the table below we will find an exact function established with the python scipy curve_fit function [2] and a simplified version of the approximation functions. These simplified functions might not be an accurate representation of the exact number of cards left, but should be a good indicator on the number of cards left for each card type.

Card type	Function	Simplified Function
4	$f(x) = 0.05x - 0.61$	$f(x) = 0.04x - 0.6$
5	$f(x) = 0.07x - 0.55$	$f(x) = 0.05x - 0.5$
6	$f(x) = 0.08x - 0.41$	$f(x) = 0.06x - 0.4$
7	$f(x) = 0.09x - 0.20$	$f(x) = 0.07x - 0.2$
8	$f(x) = 0.1x + 0.07$	$f(x) = 0.08x + 0.1$
9	$f(x) = 0.11x + 0.35$	$f(x) = 0.09x + 0.4$
10	$f(x) = 0.12x + 0.52$	$f(x) = 0.10x + 0.5$
11	$f(x) = 0.14x + 0.61$	$f(x) = 0.11x + 0.6$
12	$f(x) = 0.14x + 1.84$	$f(x) = 0.12x + 1.8$
jesters	$f(x) = 0.03x - 0.51$	$f(x) = 0.03x - 0.5$

Note that there are no approximation functions for the ranks 1, 2 and 3. This is because $a_{realistic}$ is able to memorize how many of these cards are left 3.3.1.

The general conclusion that can be drawn from the approximation functions is that for each card type (except the jesters) there is about $\frac{c_{type}}{100}\%$ left of that card type, plus or minus a small number depending on how far the card is away form the card 8 or 9.

[2] https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve_fit.html

Suppose we want to approximate how many 8's are left when there are a total of 34 cards left. We would then take the simplified approximation function for 8's and fill in the total number of cards left.

$f(34) = 0.08 * 34 + 0.07 = 2.79$, this means that when there are a total of 34 cards left, approximately 2.79 of them are an 8.

3.3.5 Approximated Play Chance

Using either the heuristics from section 3.3.3 or the approximation functions from section 3.3.4 we can use them to create a approximated version of the Play chance formula described in section 3.2.1.

$$cardsLeft(c, T) = \begin{cases} c_{left}, & \text{if } c \leq 3 \\ c_{approxLeft}, & \text{otherwise} \end{cases} \tag{5}$$

$$P_{play}(c, a, T) = \sum_{c'=c}^{12} \begin{cases} 1, & \text{if } cardsLeft(c', T) \geq a, \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

c = card we want to calculate the play chance for
a = amount of card c
T = total number of cards left

3.3.6 Approximated Round Close Chance

We can also approximate the round close chance just like we did in section 3.3.5.

$$P_{roundClose}(c, a, T) = \sum_{c'=1}^{c-1} \begin{cases} 1, & \text{if } cardsLeft(c', T) \geq a, [5] \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

c = card we want to calculate the play chance for
a = amount of card c
T = total number of cards left

We can use these probabilities to assign a value to a set of cards. This value represents how 'good' a set of cards is. The realistic player will use these values to determine the which set of cards it should play and if it should use any of the strategies mentioned in section 3.2.

3.4 Worst Card Player

A very simple, yet effective player is the worst card player ($a_{worstcard}$). This player does not count any cards and does not perform any calculations. Instead this player always plays their worst set of cards when possible (a set of rank 12 cards is worse than a set of rank 11 cards etc.). This player will mostly be used as a baseline to test the other agents against for the results section.

There are also two variations of this agent, $a_{worstcard-split}$ and the normal $a_{worstcard}$. The difference between these two agents is that $a_{worstcard-split}$ always splits their cards when possible and $a_{worstcard}$ never splits their cards.

4 Results

4.1 Player with superhuman capabilities

In this subsection we will take a look at how each separate strategy mentioned in section 3.2 performs. In all tests the player with superhuman capabilities (a_{super}) plays against 3 other players, these players always play their worst card when possible ($a_{worstcard}$).

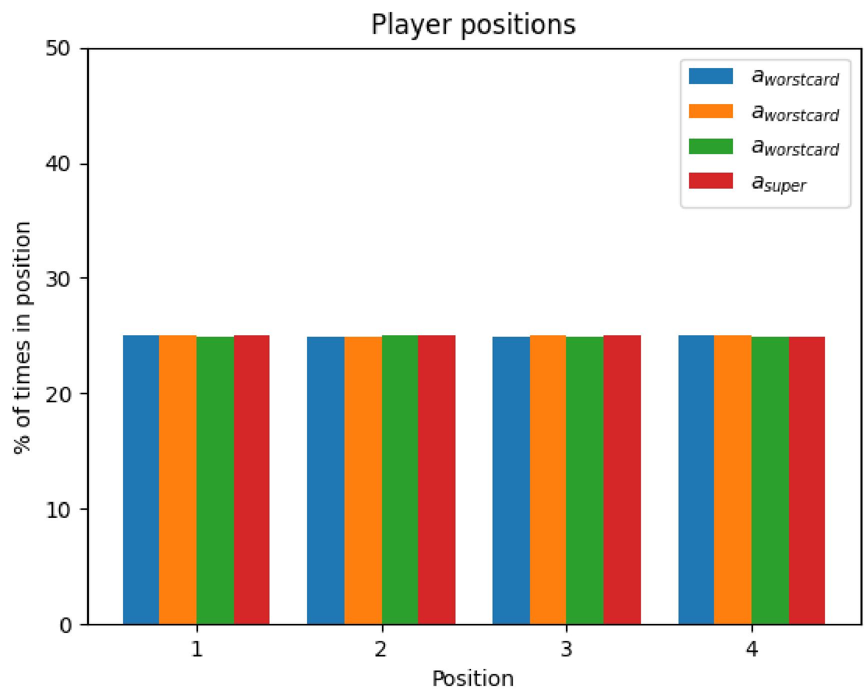


Figure 3: Position distribution of 4 $a_{worstcard}$ players playing against each other. Results obtained from 1 million simulated games.

In Figure 3, we can conclude that each of the four $a_{worstcard}$ players occupies every position an equal number of times.

4.1.1 Card Values

In this section we will take a look at how a_{super} performs against 3 $a_{worstcard}$ players when only making use of the round close chance 3.2.2 3.2.3 and the play chance 3.2.1.

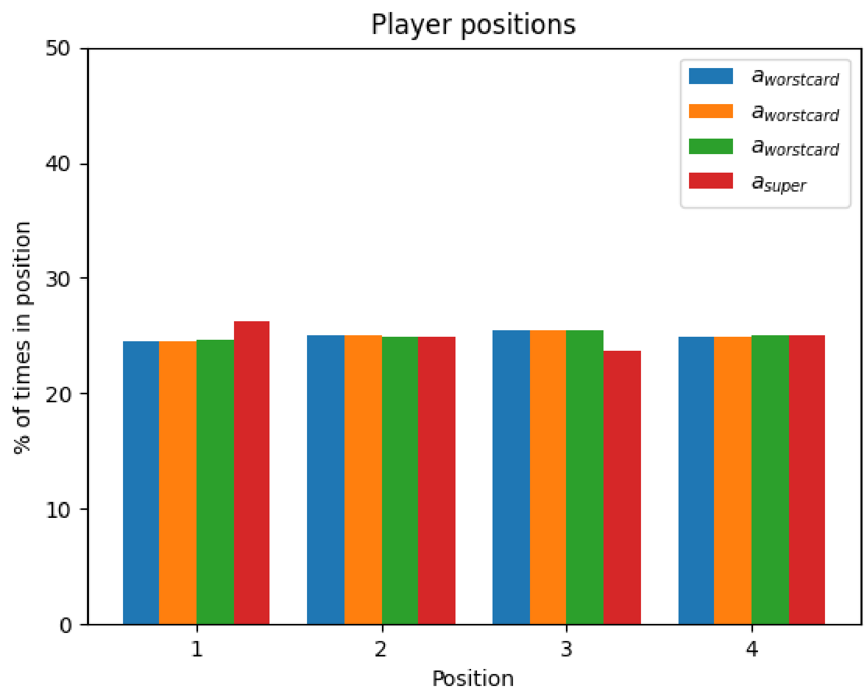


Figure 4: Position distribution of 3 $a_{worstcard}$ players and 1 a_{super} player with no strategies, playing against each other. Results obtained from 1 million simulated games.

Figure 4 shows that a_{super}, even without any strategies, gets first place about 2% more often than $a_{worstcard}$. This means it's better to play a card set of cards according to the card value than just always playing the worst card.

4.1.2 Set Chaining

In this section we will add the set chaining strategy 3.2.3 to a_{super} and see how much it improves a_{super}.

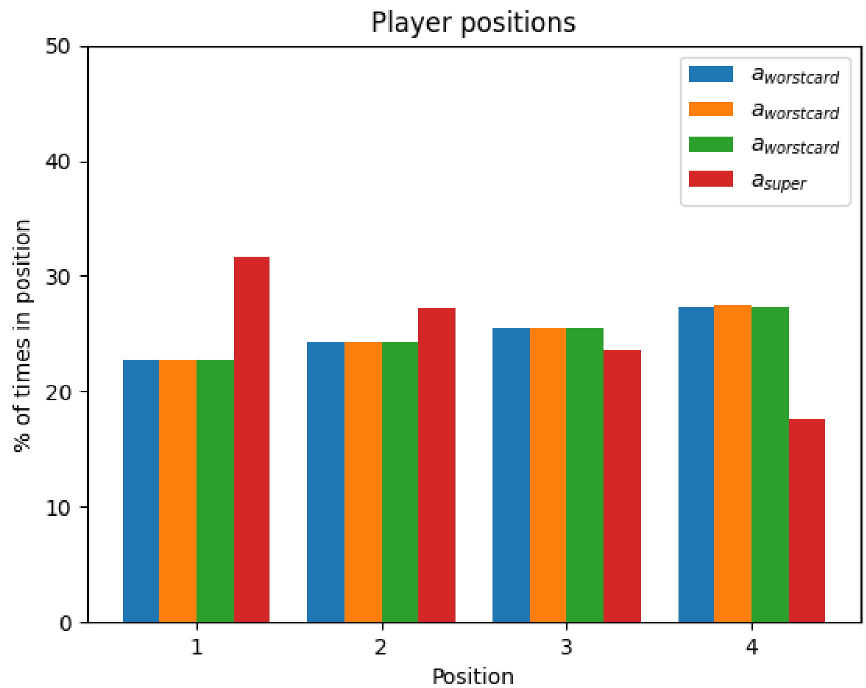


Figure 5: Position distribution of 3 $a_{worstcard}$ players and 1 a_{super} player using the chaining sets strategy 3.2.3, playing against each other. Results obtained from 1 million simulated games.

We can see in Figure 5 that adding the chaining sets strategy greatly improves a_{super}. a_{super} places first approximately 6% more, while ending up in 4th place approximately 7% less.

4.1.3 Set Closer

In this section we will take a look at how a_{super} performs using only the set closer strategy 3.2.4.

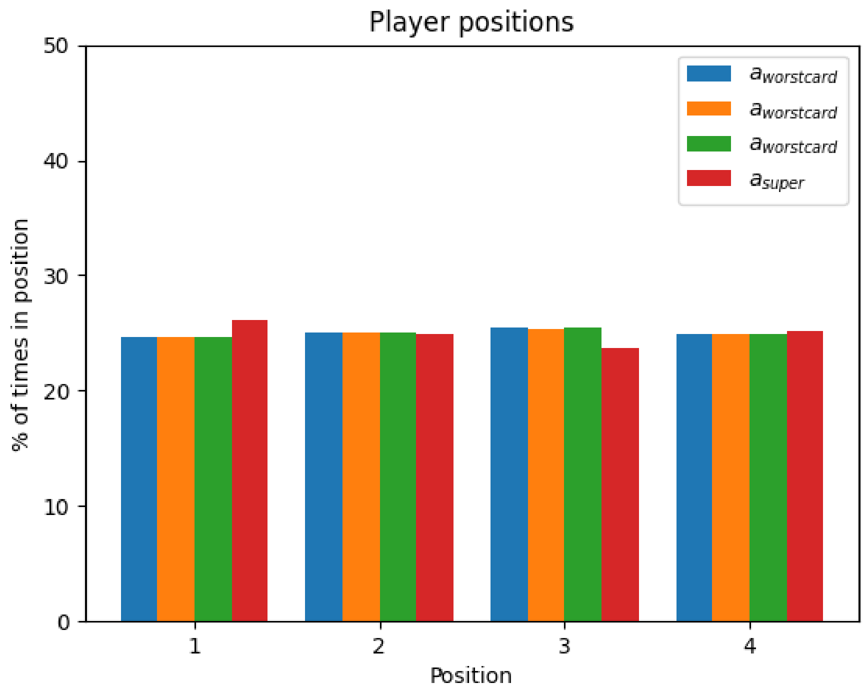


Figure 6: Position distribution of 3 $a_{worstcard}$ players and 1 a_{super} player using the set closer strategy 3.2.4 playing against each other. Results obtained from 1 million simulated games.

We can see in Figure 6 that using the set closer strategy does not give the results we might expect. There is almost no difference in the position distribution compared to Figure 4. Although there is no improvement in the performance of a_{super}, there is also no indication that a_{super} performs worse with the strategy. For this reason we let a_{super} keep the strategy since, theoretically, it should be a helpful strategy.

4.1.4 Assume a player never splits their cards

In section 3.2.5 we said that splitting cards is almost always a bad decision. In this section we will reinforce this claim by letting two $a_{worstcard}$ players who never split their cards play against two $a_{worstcard}$ player who always split their cards when possible ($a_{worstcard-split}$).

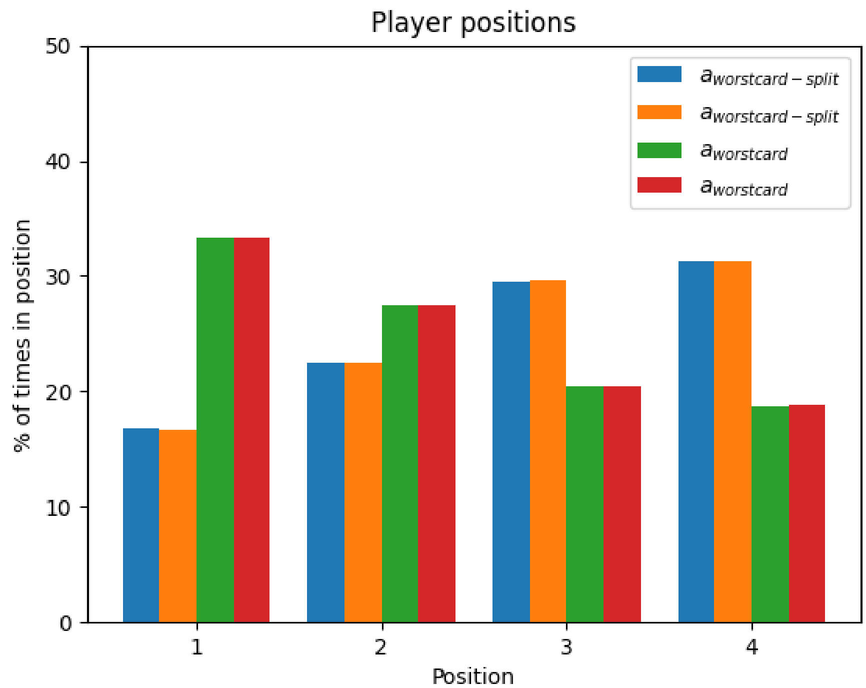


Figure 7: Position distribution of 2 $a_{worstcard-split}$ players and 2 $a_{worstcard}$ players playing against each other. Results obtained from 1 million simulated games.

We can see in Figure 7 that the $a_{worstcard-split}$ players perform way worse than the $a_{worstcard}$ players who do not split. This shows that splitting cards is usually a bad decision.

4.1.5 Superhuman player results

In this section we will take a look at how a_{super} performs when using all strategies mentioned in section 3.2 against 3 $a_{worstcard}$ players.

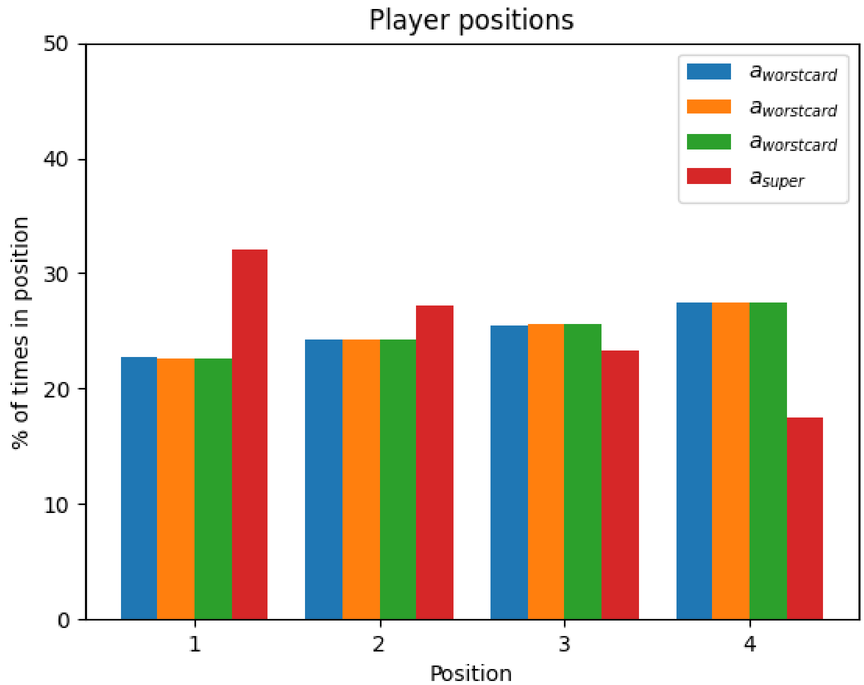


Figure 8: Position distribution of 3 $a_{worstcard}$ players and 1 a_{super} player using all strategies from 3.2, playing against each other. Results obtained from 1 million simulated games.

In Figure 8 we can see that a_{super} is better at playing the Great Dalmuti than $a_{worstcard}$. We can conclude this because, as seen in Figure 3, when all players are equally skilled at the Great Dalmuti, the distribution of their placements evens out to 25% for each position. So, if the distribution shifts to show a higher percentage of first-place finishes, it means that player is performing better than the others. The opposite is true for if the distribution shifts and shows a higher percentage of last-place finishes, this means that the player performs worse than the other players.

4.2 Realistic Player

4.2.1 Simple Heuristics

First we will take a look at a realistic player ($a_{realistic}$) using the simple heuristics stated in section 3.3.3.

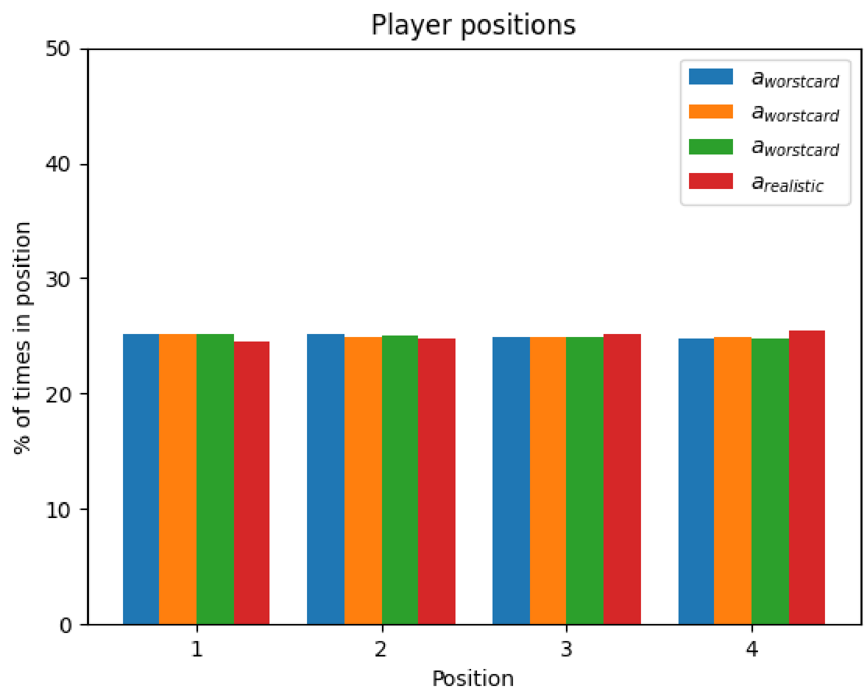


Figure 9: Position distribution of 3 $a_{worstcard}$ against 1 $a_{realistic}$ player using the simple heuristics from section 3.3.3. Results obtained from 1 millions simulated games

In Figure 9 we can see that $a_{realistic}$ actually performs worse using the heuristics than a worst card player ($a_{worstcard}$). This is surprising since we would expect that a player with more information about the current state of the game would beat a player who always plays their worst card.

4.2.2 Approximation functions

In this section we will take a look how $a_{realistic}$ performs with a more exact and simplified approximation functions as stated in section 3.3.4.

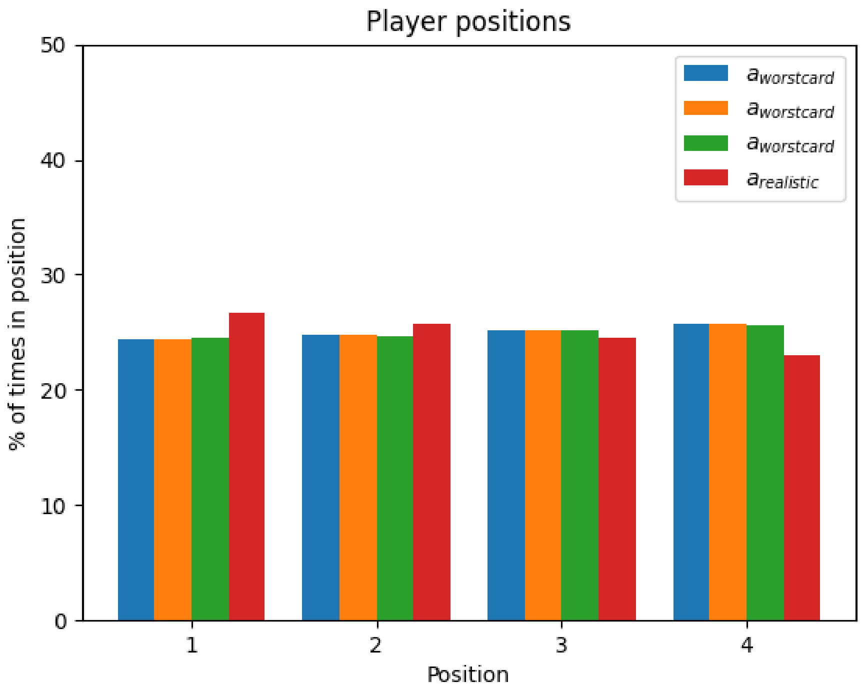


Figure 10: Position distribution of 3 $a_{worstcard}$ against 1 $a_{realistic}$ player using the exact approximation functions from section 3.3.4. Results obtained from 1 million simulated games

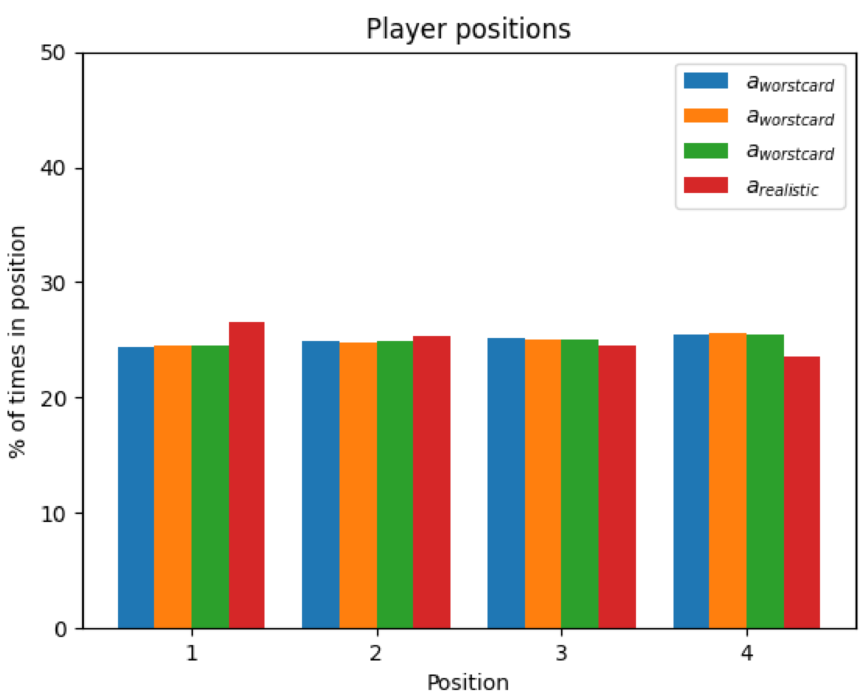


Figure 11: Position distribution of 3 $a_{worstcard}$ against 1 $a_{realistic}$ player using the simplified approximation functions from section 3.3.4. Results obtained from 1 million simulated games

When we compare Figure 10 and 11 we see that the simplified version of the approximation functions performs slightly worse. However, because the simplified functions are easier to be used as a human-like player, $a_{realistic}$ will use the simplified version of the approximation functions.

We can also see in Figure 10 and 11 that $a_{realistic}$ performs better than $a_{worstcard}$ using the approximation function. This is an improvement compared to using the simple heuristics 3.3.3 in Figure 9. This is to be expected since $a_{realistic}$ has a more accurate approximation across different stages of the game.

4.3 Realistic player compared to player with Superhuman Capabilities

In this section we will compare a Realistic player ($a_{realistic}$) from section 3.3 against a player with Superhuman capabilities (a_{super}) from section 3.2.

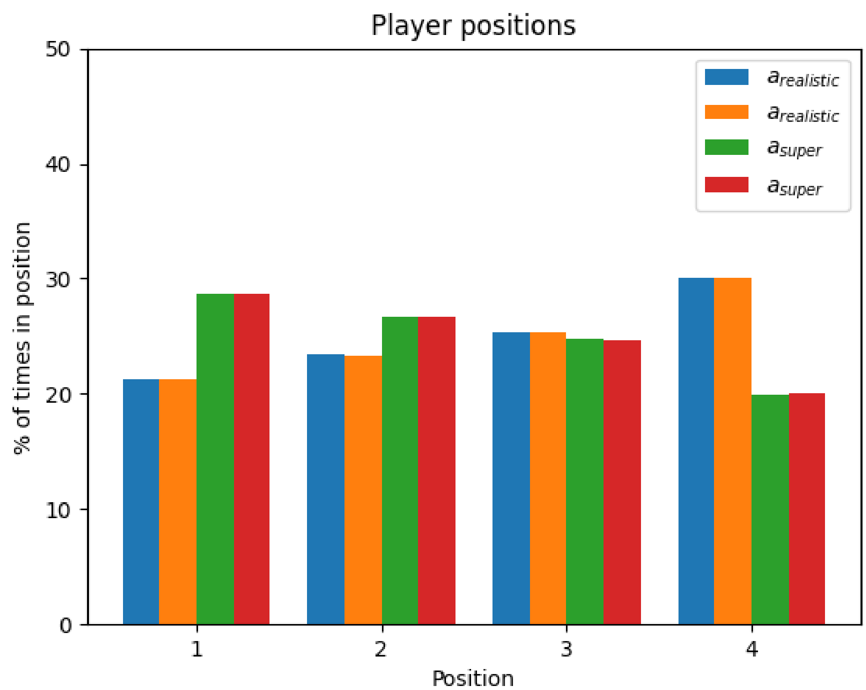


Figure 12: Position distribution from 2 $a_{realistic}$ players and 2 a_{super} players playing against each other. Results obtained from 1 million simulated games.

The results in Figure 12 are as expected. We would expect that a_{super} would beat $a_{realistic}$ because a_{super} has more memory and computing power. It is however still interesting that $a_{realistic}$ still ends up in first place approximately 20% (40%) of the games played, this suggests that a_{super} is not unbeatable by a player with less memory and computing power.

5 Discussion and Conclusion

The main objective of this thesis was to uncover or confirm strategies that work. We have seen that that the 'worst' card is not always always the card with the highest rank, like 2x 11, but could very well be the three 9's sitting in your hand. To identify the 'worst' set of cards it is important to be able to calculate or approximate the P_{play} and $P_{roundClose}$ values of a set of cards as seen in sections 3.2.1, 3.2.2, 3.3.5 and 3.3.6. These values are best used to determine when you can use the Chaining sets strategy as seen in section 3.2.3. This strategy by far improves the chances of ending up in first place as seen in section 4.1.2.

In Figure 6 we see that there is almost no difference in the position distribution compared to Figure 4 This could be due to two reasons:

- The use case for the strategy is very rare, meaning that we don't often have the chance to make use of the strategy.
- The strategy does not work well against $a_{worstcard}$

Both of these reasons are very likely to be true.

We have also seen that memorizing certain cards and approximating P_{play} and $P_{roundClose}$ has a positive impact on the position distribution as proven in section 4.2.2. This is a good sign for human players, since they are also able to apply the strategies that player with superhuman capabilities uses (a_{super}).

A very interesting result was that the player who always played their worst set of cards ($a_{worstcard}$) is a surprisingly good player. This is probably because the $a_{worstcard}$ follows the basic principles of the game the Great Dalmuti, namely, getting rid of your cards as fast as possible. This means that if you do not want to play to the best of your capabilities, but still want to have a good chance of becoming first, just always playing your worst set of cards is a good strategy.
In summary, this research has been very beneficial when it comes to confirming and discovering strategies. Human players can take notes from the realistic player ($a_{realistic}$) and apply the strategies that $a_{realistic}$ to their own game play.

6 Future work

In this thesis we left out passing this is because when deciding if you should pass you ask the question: are my chances of becoming first higher if I pass? This is a difficult question to answer because we do not know whether passing was the right play immediately, only after the game has finished. Passing is also dependent on which actions our opponents take, our own cards, total cards left, how many players have already passed, the cards of our opponents, etc. Because it is dependent on so many factors, it is hard to say when to pass. I do suspect however that having the ability to know when to pass as a player could greatly improve the position distribution of a player.

Future work could be dedicated to implementing this passing functionality. The ability to pass would probably be implemented in some sort of deep-reinforcement learning algorithm since it is dependent on many factors.

There is also improvement to be made on the realistic player $a_{realistic}$. Something like theory of mind (ToM) [FF05] could help the realistic player make decisions that take the possible actions of other players in to consideration. This should have a positive impact on the position distribution of the player.

In this thesis we let the players make decisions solely based on the values of their cards determined from the formulas in sections 3.2.1 and 3.2.2 or 3.3.5 and 3.3.6. Future work could try to create a player that uses reinforcement learning to determine which action it should take. This player should be much more flexible and adapt to the play style of their opponents.

Overall, I hope that in the future this work can be expanded on and that new promising strategies get uncovered for human players that they can use to improve their chances of becoming first in their games.

References

[CCM16] Mustapha Chekaf, Nelson Cowan, and Fabien Mathy. Chunk formation in immediate memory and how it relates to data compression, Oct 2016.

[Cow10] Nelson Cowan. The magical mystery four: How is working memory capacity limited, and why? *Current Directions in Psychological Science*, 19(1):51–57, 2010. PMID: 20445769.

[FF05] Chris Frith and Uta Frith. Theory of mind. *Current biology*, 15(17):R644–R645, 2005.

[ICDEC+11] Javier Insa-Cabrera, David L. Dowe, Sergio España-Cubillo, M. Victoria Hernández-Lloreda, and José Hernández-Orallo. Comparing humans and ai agents. In Jürgen Schmidhuber, Kristinn R. Thórisson, and Moshe Looks, editors, *Artificial General Intelligence*, pages 122–132, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[OSV] Frans Oliehoek, Matthijs Spaan, and Nikos Vlassis. Best-response play in partially observable card games.

[WD92] Christopher J. C. H. Watkins and Peter Dayan. Q-learning - machine learning, May 1992.

[WZZ+16] Fei-Yue Wang, Jun Jason Zhang, Xinhu Zheng, Xiao Wang, Yong Yuan, Xiaoxiao Dai, Jie Zhang, and Liuqing Yang. Where does alphago go: from church-turing thesis to alphago thesis and beyond. *IEEE/CAA Journal of Automatica Sinica*, 3(2):113–120, 2016.