The Hardness Analysis of FreeCell Games

Michael W.H. Chen
10411941

Bachelor thesis
Credits: 18 EC

Bachelor Opleiding Kunstmatige Intelligentie

University of Amsterdam
Faculty of Science
Science Park 904
1098 XH Amsterdam

Supervisor
Daan van den Berg

Faculteit der Natuurwetenschappen, Wiskunde en Informatica
University of Amsterdam
Science Park 904
1098 XH Amsterdam

June 26th, 2015

UNIVERSITY OF AMSTERDAM

Abstract

Hardness is a crucial factor for people to find enjoyment in (puzzle) games. Games that are not suitable for players based on their skill, could change enjoyment, for instance into frustration for the player, which is not desirable. Furthermore, the majority of solvers in NP-Complete problem areas utilises brute-force algorithms or simple heuristics, which is not ideal given the complexity. This thesis presents an attempt to establish a model that could a priori determine the hardness of FreeCell games by analysing instances of the game. The four examined game characteristics were the placements of the aces, the placements of the kings, games sorted by colour and the cards blocking the aces. As all four characteristics affected the solution length, the number of checked states and the run time of the solver, which were the three hardness indicators for this research, a model that allocates hardness scores to games based on the characteristics was developed. The performance of the model was evaluated by examining the 'Pearon's Correlation Coefficient *r*'. As a result, the model achieved the highest correlation with the solution length indicator, with a r-value of 0.439, followed by a r-value of 0.366 for both the number of checked states and the run time indicators. Therefore, the overall performance of the developed model was weak to moderate, implying that the developed model may not be an acceptable hardness indicator for FreeCell games.

Contents

1 Introduction

This section gives a brief introduction to the subject FreeCell, explains the problem areas and poses the research question that will be investigated with the corresponding hypothesis.

1.1 Problem Definition

Difficulty is a crucial factor for people to have enjoyment in puzzle games [McFarlane et al., 2002]. Games that are too simple tend to become boring, while games that are too difficult could cause stress and frustration for the player. In exceptional situations, it may result in people quiting the game, due to the fact that the game is extremely hard or extremely easy. In other words, the hardness outweighs the entertaining aspect in these particular situations, which is not the reason of playing games. On the contrary, games are usually played to relieve people from stress. This feature would disappear when games are simply too hard to play. Hence, a hardness measurement for puzzle games is desirable.

This thesis focuses on the card game FreeCell. Since FreeCell is a complex and challenging game, it would be difficult to indicate the hardness, based on the initial state of the game. Furthermore, the majority of FreeCell solvers utilises brute-force algorithms or simple heuristics, which is not ideal given the complexity of FreeCell. As the generalised version (FreeCell games with $4\ x\ n$ cards) of FreeCell is NP-Complete, no more efficient algorithm than a brute-force method exists that can solve random FreeCell configurations [Helmert, 2003]. If the characteristics that determine the hardness of the game are recognised, most likely better solvers could be implemented that are based, for instance on intelligent heuristics, rather than brute-force methods and simple heuristics. As a result, it might accelerate the overall solving process. Additionally, it could provide more insight in differences between instances, regarding the hardness of other computational problems in general.

1.2 The card game FreeCell

FreeCell is a solitaire or patience card game, which implies that the game is designed to be played by one player at a time. This game makes use of a standard deck of 52 cards (fig. 1). The figure displays an example of an initial state of the game. The game consists of four *foundations* that are empty in the initial state (the four top-right cells) and four *free cells* that function as a storage area with a capacity of one card for each cell, whose purpose is to aid with moving the cards (the four top-left cells). The deck is randomly divided into eight different piles of cards, which are called *cascades*. Each turn, the player can only move one exposed card, either from the free cells or cascades. Here the definition of ‘exposed’ is that there are no other cards blocking the selected card. The selected card could only be placed on top of another card, if the selected card is one rank lower and has a

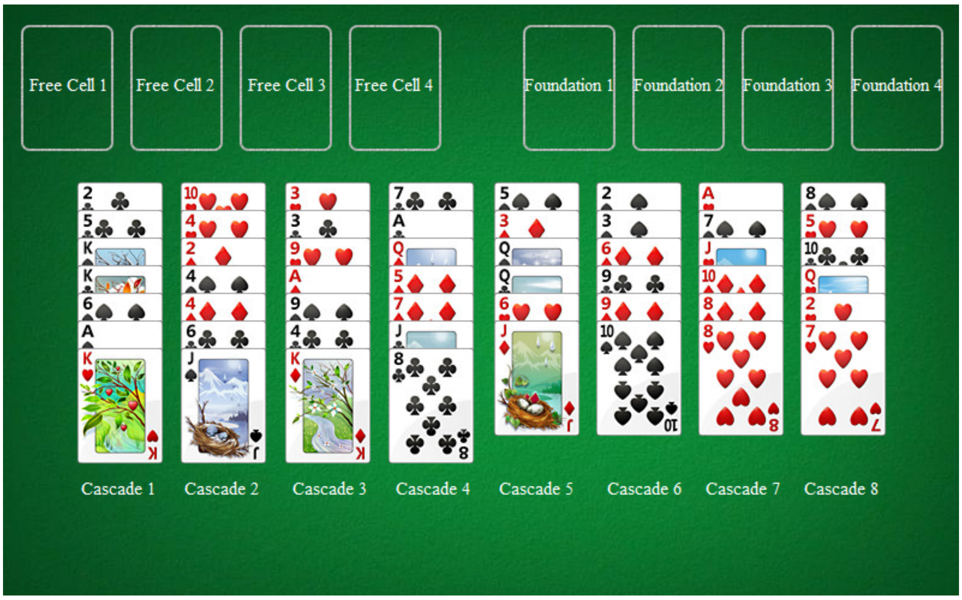

Figure 1: A version of FreeCell (Microsoft). The purpose of the game is to move all cards into the four foundations in ascending order, sorted by colour. Each turn, the player can select one exposed card, including cards in the free cells. The selected card can be placed on top of other cards, only when the card rank differences is equal to one and the colour is different.

different colour (i.e. red or black). Additionally, it is possible to perform a ‘super move’, where a sequence of cards is moved, only if it is possible to relocate the whole sequence by moving one card at a time, using free cells and/or cascades. This move is the only method to move multiple cards in one turn. The objective of the game is to move all cards in the cascades into the four foundations, in ascending order (starting with the aces and ending with the kings), sorted by suit.

1.3 Motivation

As it has been previously outlined, given the fact that FreeCell is a highly challenging and complex game, it would be useful to derive a model that could a priori determine the hardness of games, as it is commonly hard to indicate, whether a game is difficult just by observing the initial state of the game, as many factors could play a role. Thus, this thesis attempts to define a measure of hardness a priori for game instances of the card game FreeCell.

By discovering numerous game characteristics that could potentially determine the hardness of FreeCell games, a predictive model could be developed that predicts the hardness of FreeCell games, or at least give a hardness indication with reasonable accuracy, without actually attempting to solve the game. As a result, more suitable games could be generated based on the skills of the player, which causes the game to remain enjoyable and challenging. Additionally, more accurate health diagnosis could be achieved in the health care, when puzzle games are used to determine the cognitive abilities of people [Jimison et al., 2003, 2004], described in section 2 'Related Work'. Furthermore, the model could be used to improve algorithms of FreeCell solvers, for instance using the model as a tuning tool. Lastly, the attained game characteristics and the developed model could both function as a foundation for better understanding of the difficulty of FreeCell games in future work.

1.4 Research Question & Hypothesis

The first thing that needed to be done, was to examine which game characteristics affect the hardness of FreeCell games. Therefore, this thesis attempted to answer the following research question:

Which game characteristics affect the hardness of FreeCell games and what type of FreeCell games are hard to solve?

After becoming familiar with the game rules and playing some FreeCell myself, it was noticeable that the placements of the aces and kings could have a great impact on the hardness. For instance, if the game started with the aces located near the bottom of the cascades, generally it would be harder to reach the aces. This is not favourable, since the foundations must be started with the aces. The same applies when the kings are placed near the top of the cascades, preventing all other cards with lower ranks to be moved into the foundations. Additionally, when cards with high ranks are blocking the aces, it would likewise be hard to clear the aces. Therefore the initial assumption would be that the placements of the aces and the kings and the ranks of the cards blocking the aces are potential game characteristics that affect the hardness of FreeCell games.

The remaining parts of the thesis proceed as follows: section 2 highlights other work that has been done related to FreeCell. Section 3 describes the methodological approach that has been executed during this research in detail, with the purpose to develop a model by answering the research question. Following that, section 4 displays the results that were achieved. In section 5, the evaluation of the developed model is explained. Subsequently, section 6 consists of concluding notes regarding the evaluation and results. Lastly, section 7 discusses some flaws and areas for improvements, ending with future research directions.

2 Related work

FreeCell possesses several traits that are appealing for researchers in the domain of artificial intelligence. Firstly, FreeCell is generally considered a complex and challenging puzzle game for humans. Artificial intelligent researchers are especially interested in how humans solve these particular types of games is, as they seek to produce the same results with machines. Secondly, there are no chances involved, when one is playing the game, since every performed move is deterministic, due to the fact that all cards are faced up during the entire game. As a result, several investigations have been conducted regarding FreeCell solvers.

Several solvers have tried to solve all 32000 games of the Microsoft FreeCell package, known as the 'Microsoft 32000'. Later on, it was discovered that game 11982 had no solution [Heineman, Heineman et al., 2008]. Nevertheless, solving the remaining games remained hard [Benbassat et al., 2013].

The most successful FreeCell solver was developed by Elyasaf et al. [2012]. By solving 99.65% of the standard Microsoft 32K problem set in reasonable time, it was most likely one of the best existing solvers. Although, the percentage of the solved games may be high, the results are still based on games of one problem set. Therefore, there is still much room for improvement regarding the FreeCell solvers.

Additionally, thoughts about incorporating AI-elements into the solvers were discussed by C. Chan [Chan]. Artificial intelligent algorithms, such as Bayesian Learning, Decision Trees and Neural Networks have shed some light for possible improvements of FreeCell solvers in general. Unfortunately, since most AI-research was focused on multiplayer games rather than single player games, insufficient work has been published regarding applications of artificial intelligence in solvers of FreeCell or any other solitaire-like games.

Given the strategic and challenging environment provided by FreeCell games, researchers have speculated, utilising FreeCell to help the elderly and patients with cognitive impairments [Jimison et al., 2003]. In recent years, assistance for the elderly in maintaining their independence and quality of life has become an important issue, due to the increase of the ageing population. Nowadays, more people reach an older age compared to the previous generations. A common phenomenon that occurs among the elderly is the decrease in cognitive ability. By detecting early decreases in cognitive ability, possibly more effective clinical interventions could be utilised, for instance to counteract dementia.

As a result, a *research* version of FreeCell was developed to evaluate the cognitive abilities of the elderly [Jimison et al., 2004]. An acceptable hardness rating system to determine the hardness of FreeCell games is crucial. Otherwise imprecise results could be produced, causing unnecessary harm, such as incorrect diagnoses, which may lead to incorrect medical treatments. The research version of FreeCell has emphasised the number of cards to be moved, in order to solve the game as the indicator for hardness of FreeCell games. In their case, the hardness measure changes during the game, as it corresponds to the remaining moves the user needs to perform in order to solve the game.

This is in some ways different from the hardness indicators that were used in this research. Firstly, game characteristics that potentially affected the hardness of FreeCell games were examined. This was accomplished by considering the total number of checked states and the run time of the solver as indicators for hardness of FreeCell games, along with the total number of moves to reach the solution (i.e. the solution length). Subsequently, a hardness measurement (i.e. the model) was allocated to each game, based on the characteristics that may affect the hardness of FreeCell games, which ensures that the hardness is known before playing the game. Additionally, it makes it possible to generate games of a certain hardness, which may be beneficial for aiding people with cognitive impairments, explained in Jimison et al. [2003] and Jimison et al. [2004]. Moreover, some attempts have been made to automatically rate Sudoku puzzles in a human perspective [Wang et al., 2012]. By measuring the hardness of logic puzzle games based on the solving path of the solver and focusing on features, such as the number of back traces, human-perceived difficulty could be estimated. This method could also function as a framework for other puzzle games, such as FreeCell.

3 Method

This section describes the methodological approach that was taken during this research. It could be divided into four main tasks: defining several game characteristics to investigate, implementing a game generator to generate different types of games, analysing the solutions of the generated games and establishing a model to rate the hardness.

3.1 Defining Several game characteristics

3.1.1 Definition of hardness

The hardness of each FreeCell game is based on three indicators. The first indicator is the solution length. Generally, it could be said that the longer the solution length, the harder the game might be, because more card moves need to be utilised to reach the solution.

The second indicator corresponds with the number of checked states. The states in question corresponds with positions in the state-space of the game, as each game with their corresponding moves could be visualised with a graph. In other words, the states correspond with the game states. If the solver has performed a move, the game state changes, which results in a new state. So, the total number of checked states of the solver could be considered as the total number of steps the solver needed to take to solve a game.

This roughly corresponds with the time spent by the solver, because a longer run time implies that most likely more states are inspected [ShlomiFish]. Hence, the run time in milliseconds of the solver will be the third indicator for hardness. So, if more states needed to be checked and thus results in a longer run time, it most likely indicates that the game was harder to solve.

3.1.2 Game Characteristics

It was difficult to derive some meaningful characteristics. Therefore some research has been done to obtain a better understanding of the game, such as understanding the game rules, reading relevant literature and the most effective one is spending time actually playing FreeCell. After this, some assumptions regarding game characteristics that potentially affect the hardness were derived, as described in section 1.4.

During this research a total of four game characteristics were investigated. The first characteristic is the placement of the aces, the second characteristic is the placement of the kings, the third characteristic is the rank of the cards that are blocking the aces and the last characteristic is the colour ordering of the cards. The colour sorting characteristic was also taken into account, since one of the rules states that cards can only be placed on top another card, if the card has a different colour. Therefore, it should be interesting to examine games sorted by colour. Section 3.2 provides more information and details regarding the game characteristics.

3.2 Different types of games

To examine which of the characteristics have an effect on the hardness, approximately 30000 games have been generated for each characteristic.

It is relevant to note that all games were checked for uniqueness, since it could be possible that equal games were generated. This is necessary to guarantee reliable results.

Below, all different types of games that were generated during this research will be explained in detail.

3.2.1 Random Games

Random games are necessary to investigate, whether a certain game characteristic affects the hardness of FreeCell games. *Random* means that all cards are placed randomly to generate the initial state of the game. The average values of the three hardness indicators (i.e. solution length, number of checked states and run time) of these random games are considered as the baseline. So if different values were found with games that were generated with a certain characteristic, compared with the values of the baseline, it could be concluded that the concerning characteristic potentially affects the hardness of FreeCell games.

```
8S QS TD JS 6D 2H 6S 9S
4H 2D 9C 2C 8D 5D 4D 6H
7D TS JC 5C AC TH 7H 4C
QH AS 6C 9D TC KH 3H JH
KC 7S 8C AD 3C KS 3S QD
7C 5H JD 9H 4S AH 3D 5S
QC 2S 8H KD
```

Figure 2: Example of a randomly generated game.

Figure 2 displays an example of a randomly generated game. The symbols `T,J,Q,K` are abbreviations of the ranks *Ten, Jack, Queen* and *King*. The symbols `S,H,D,C` are abbreviations of the suits *Spades, Hearts, Diamonds* and *Clubs*. For clarification, the exposed cards in this example are `QC,2S,8H,KD,4S,AH, 3D,5S`.

3.2.2 Characteristic 1 & 2: Placements of the aces and kings

Different types of games were generated based on the placements of the aces and kings. The placements are designated using the variable *depth*. The variable *depth* is defined as the number of cards that are located on top of the aces and the kings in the initial state of the game. Hence, the variable depth can range from 0 to 6. To emphasise, *all remaining cards were placed randomly. Only the aces and kings were placed according to the value of the variable depth.* In this case, two types of games were generated: games where the aces were placed at depth 0 and the kings at depth 5 or 6 of the cascades (i.e. depth 5 or 6, because the number of cards per cascades consists of 6 or 7 cards) and games where the kings were placed at depth 0 and the aces at depth 5 or 6 of the cascades. Two examples are displayed in figure 3 to clarify these two types of games.

3S	**KH**	4C	**KC**	**KS**	**KD**	TS	3H
5C	QD	9S	TH	4H	6S	7H	7D
QC	QS	QH	3C	9H	5S	9D	6C
3D	2S	JC	4D	8S	JS	5H	7C
2H	8H	TD	6H	5D	7S	JH	2D
JD	8C	6D	2C	4S	**AD**	9C	TC
AS	**AC**	8D	**AH**				

AS	9D	**AD**	**AH**	8D	**AC**	6H	TS
9S	8H	7D	QC	5C	JH	QS	8S
3D	4C	9C	2C	8C	9H	3C	7H
3H	5H	4H	2H	5S	2S	QH	6D
7C	JD	QD	JS	2D	6C	4S	TD
6S	4D	3S	5D	**KS**	**KC**	TC	**KH**
7S	JC	TH	**KD**				

Figure 3: The left game is an example, where all four aces are placed at depth 0 and the kings placed at depth 5 or 6 of the cascades. The right game is an example, where all four aces are placed at depth 5 or 6 and the kings placed at depth 0.

Additionally, to display the impact of the placements of the aces and kings on the three hardness indicators separately, extra games were generated with varying depth, from zero to six. So, there were games generated where only the aces were placed with a varying depth and games where only the kings were placed with a varying depth (i.e. a total of 14 different types of games). Again for the illustration, two examples are given in figure 4.

KH	5H	6H	7C	2D	3C	2C	QC
9C	KC	9H	8S	KS	4C	5D	7S
QD	8D	JC	TS	JD	5C	JH	QS
3D	6C	QH	4H	3H	8C	6S	TC
5S	TD	2H	7D	**AC**	**AH**	9D	9S
AS	6D	2S	**AD**	KD	8H	4D	TH
7H	JS	3S	4S				

KS	**KH**	**KD**	**KC**	8C	4S	4H	6C
4D	4C	3D	AD	TD	3C	5S	8D
QC	JD	JH	7H	AS	5H	3H	2S
AC	JS	QS	JC	7C	9S	AH	9H
QD	7S	6S	8H	2D	TS	8S	2C
TH	6D	6H	2H	5C	9C	9D	TC
5D	7D	3S	QH				

Figure 4: The game on the left had the aces placed at depth 1 (one card on top of the aces) and the kings were placed randomly. The game on the right had the kings placed at depth 6 (six cards on top of the kings) and the aces were placed randomly.

3.2.3 Characteristic 3: Sorted by colour

The next examined game characteristic was the colours. Since one of the game rules declares that a card can only be placed on top of another card, when the selected card is one rank lower and has a different *colour*, it would be quite interesting to examine, what kind of effect the colours have on the hardness, when the games are sorted by colour. For this reason, the same types of games were generated as described in section 3.2.2, but sorted by colour. Note that the game only consists of two colours, with the diamonds and hearts belonging to the *red* cards and the spades and clubs belonging to the *black* cards. Therefore, games sorted by colour imply that every cascade contains either only red cards or only black cards. Two examples of games sorted by colour are displayed in figure 5 for clarification.

8D KH 6S 7S QC 4H 4D 7C	3C TD AH 4S 3H 7H 8S 3S
AH TD 8C 9S TC QH 2D 4S	JS JH 2D 7C 6H TH QS 2C
3H 5D JS JC 8S QD JD KS	9S 5D 4D 2S 8D 9D 5C 4C
TH AD 6C KC 3S 9H 2H 4C	8C 2H KD KS QD 5H JC KC
7H 6H 9C 2S AC 8H JH 5C	9C AD 6D 7S 8H 7D AC TC
7D 6D AS QS TS 5H 9D 3C	6S QH 9H QC 3D KH TS AS
KD 3D 5S 2C	6C JD 4H 5S

Figure 5: Two example games generated with the colour sorting characteristic. Every cascade contains either only red cards or only black cards.

3.2.4 Characteristic 4: Blocking cards

The last investigated characteristic was the rank of the cards blocking the four aces. Throughout the thesis, these cards are named as the *blocking cards*. For instance, blocking card `Q` means that the four queens are placed on top of the four aces, thus ‘blocking’ the aces. There is a total of twelve different types of games generated, since there are twelve possible card ranks that could block all four aces. Two examples are displayed in figure 6 to demonstrate this characteristic.

JH 4S QC 3C 9H AH KH KD	6S 7S 2S 6D JS QH 8C AD
KS 8H 4H 8S 9D **TH** 8C 3D	5S 8D AC 6C 7C JD QC **KD**
2S JD AS 5H QS 4D QH KC	9H 2D **KS** 6H 3S 4C 3D 4D
3S 6C **TS** 5C AD JC AC 9C	4S 5D QD 9D 4H 3C 8S JH
2D JS QD 7D **TD** 6H **TC** 5D	AS TD TC 7D 5C 3H QS AH
5S 7S 7H 3H 8D 6S 7C 9S	**KH** JC 9S 8H 7H TS 5H **KC**
6D 4C 2C 2H	2C TH 9C 2H

Figure 6: The game on the left represents a game, where the aces are blocked by the tens (blocking card `T`) and the game on the right represents a game, where the aces are being blocked by the kings (blocking card `K`).

3.3 Solving the games

3.3.1 Shlomifish FreeCell Solver

The Shlomifish FreeCell Solver was used during this research [Fish, 2015a,b] to solve each generated game, with the purpose to provide the relevant values for the analyses regarding the three hardness indicators explained in section 3.1.1.

Different heuristics were tested on a small test set to single out the best heuristic to use for this research, as the solver supports multiple heuristics. The first test was using the default settings, consisting of the 'Depth-First Search' method with the default test order of `[01][23456789]`. Table 1 displays the actions belonging to the numbers.

	Action
0	put top stack cards in the foundations.
1	put freecell cards in the foundations.
2	put freecell cards on top of stacks.
3	put non-top stack cards in the foundations.
4	move stack cards to different stacks.
5	move stack cards to a parent card on the same stack.
6	move sequences of cards onto free stacks.
7	put freecell cards on empty stacks.
8	move cards to a different parent.
9	empty an entire stack into the freecells.

Table 1: Different actions that could be performed by the solver according to the order. The default order is `[01][23456789]`. The order of the actions could be changed.

The run time of the solver was quite diverse, as Depth-First Search operates by visiting the deepest nodes first, whenever the algorithm is given a choice of continuing the search from several nodes [Bratko, 2012, p. 265-270]. Therefore, the run time would not be a good hardness indicator. Additionally, the solution lengths are most likely not optimal, given the nature of the Depth-First Search heuristic, which lowers the value of the solution length hardness indicator.

Every game has an optimal (shortest) solution length. So the solution length indicator would be more meaningful if it was produced by a solver that yields shorter solutions on average. The next examined heuristic was the 'Best-First Search' method. Due to the fact that this algorithm computes a heuristic estimate for each candidate in his candidate path and chooses the best candidate according to this estimate for expansion, shorter solutions are assumed [Bratko, 2012, p. 280-289]. The test displayed that the average solution length was shorter, compared with the solution lengths produced by the Depth-First Search method. However, the run time to solve each game was considerably longer. As the Best-First Search method would seem a better choice than the Depth-First Search, due to the shorter solutions, it is still not applicable given the huge amount of games that were generated.

Fortunately, the solver supports another option to solve FreeCell games, which are called *presets*. A preset could be considered as a set of command line arguments to be processed as a command line. So a set of commands will be executed one at a time, may get suspended or resumed based on the given parameters. The solver is done executing, when a solution is reached.

During this research, the preset with a extraordinary name *Video Editing* was used. This preset utilises *flares*, which are various alternative scans (i.e. executions of commands), that are ran one after another, attempting to solve the game with different settings for the parameters. The FreeCell solver chooses the one with the shortest solution out of all flares that were successful in solving the game. According to an initial survey, the solutions produced by the *Video Editing* preset were fairly short solutions and the run times were likewise acceptable. Therefore, this preset was suitable for this research.

3.3.2 Game Solutions

To illustrate how every solution appears in a txt-file, an example game is displayed in figure 7.

```
7S 9S KD QC 5S 5C 9D 5H
TC 4D AC 6C 4S JD 5D AS
7C TD 2S 7D 2D 3C 8C 8S
2C KH TH 4H 4C QH 7H KC
3H KS 9H 2H 6S 6D 8D 3S
9C 3D 8H QS AD QD JS JC
JH AH 6H TS
```

Figure 7: An example game to illustrate the solution produced by the solver. See figure 8

The corresponding solution produced by the Shlomifish FreeCell Solver, using the preset *Video Editing* is showed in figure 8.

```
-=-=-=-=-=-=-=-=-=-=-=-

5h 2h 3a 3b 3c 37 3d 3h d2 41
4d 4h d3 13 1d 1h 1h 4h d7 b7
a1 86 8d 8b 8a 8h 2h dh 8h 1h
5d 58 5h 5h 2h 5h dh 1d 15 1h
ah 62 6a 61 6h 8h 68 6h 46 4h
dh 58 84 c4 71 7d 7h 67 1h d1
4h b5 15 7c 7h 2d 2b 28 26 21
2h 7h ah ch 2h 5h 3h 7h 1h 5h
4h 5h 3h dh 4h 5h 3h 8h bh 4h
5h 3h 6h 5h

This game is solveable.
Total number of states checked is 74152.
This scan generated 143061 states.
```

Figure 8: The solution of the game displayed in figure 7, produced by the Shlomifish FreeCell Solver, using the preset *Video Editing*. The solution is represented in the standard FreeCell notation. Every pair corresponds to one move.

The solution of the game is represented in the standard FreeCell notation. The letters `a,b,c,d` correspond to the free cells, the letter `h` corresponds with the foundations and the numbers `1` to `8` correspond with the cascades. The first symbol of every pair corresponds with the location of the selected card and the second symbol corresponds with the destination. For instance, the first move of this solution is `5h`, which implies that the exposed card at cascade 5 was moved to the foundation. Looking back at the example game, the card `AD` will be placed in one of the foundations. The second move `2h` implies that the exposed card at cascade 2 (in this case `AH`) was moved to the foundation. In other words, every pair corresponds with one move performed by the solver. So the total number of pairs, corresponds with the solution length. The total number of checked states was already calculated by the solver and outputted in the solution file. The run times were calculated separately.

4 Results

This section gives an overview of the results that were achieved during this research. For every solved game, the corresponding solution was analysed in order to obtain several meaningful values for the three hardness indicators.

As it was previously outlined in section 3.2, different types of games were generated for each game characteristic. Figure 9 illustrates the influences of the placements of the aces and kings and the colour sorting characteristic on the solution length, the number of checked states and the run time of the solver. Recall that *depth* was defined as the number of cards placed on top of the aces and kings. The average values for the three hardness indicators is calculated based on 2000 games for every depth. All three figures display the same patterns, as all values for the three hardness indicators increases, the lower the aces are located in the cascades (i.e. higher depth value). This seems logical, since the foundations can only be started with the aces and more cards on top of the aces obstruct the aces to be moved into the foundations.

The exact opposite results applies for the kings, as the values for the hardness indicators decreases, the larger the depth value for the kings. Obviously, kings placed relatively high in the cascades is not desirable, since the foundations must be ended with the kings, which withhold lower card ranks to be placed in favourable positions, when the kings are located high in the cascades (low depth value).

In addition, it is interesting to note the values for the colour sorted games are significantly higher for the number of checked states and the run time, than the games not sorted by colour. The plots seem to indicate that the colour sorting characteristic has a much greater impact on the number of checked states and on the run time, when compared with the solution length.

Figure 10 displays the impact of the blocking cards characteristic on the three hardness indicators. All three figures indicate an increase in average value of the solution length, the number of checked states and the run time, the higher the rank of the blocking cards. However, the effect of the blocking cards characteristic on the three hardness indicators, seems to be less impactful, compared to the previous three characteristics displayed in figure 9, as some parts of the plots are constant. For example, the card ranks 9, 10 and 11 resulted in the same average solution length equal to 96. Furthermore, the variation/range of the values for the hardness indicators in figure 10 is less than the variation/range of the values in figure 9. For instance, the solution length in figure 9 ranges from circa 80 to 110, while the solution length in figure 10 ranges from 88 to 97.

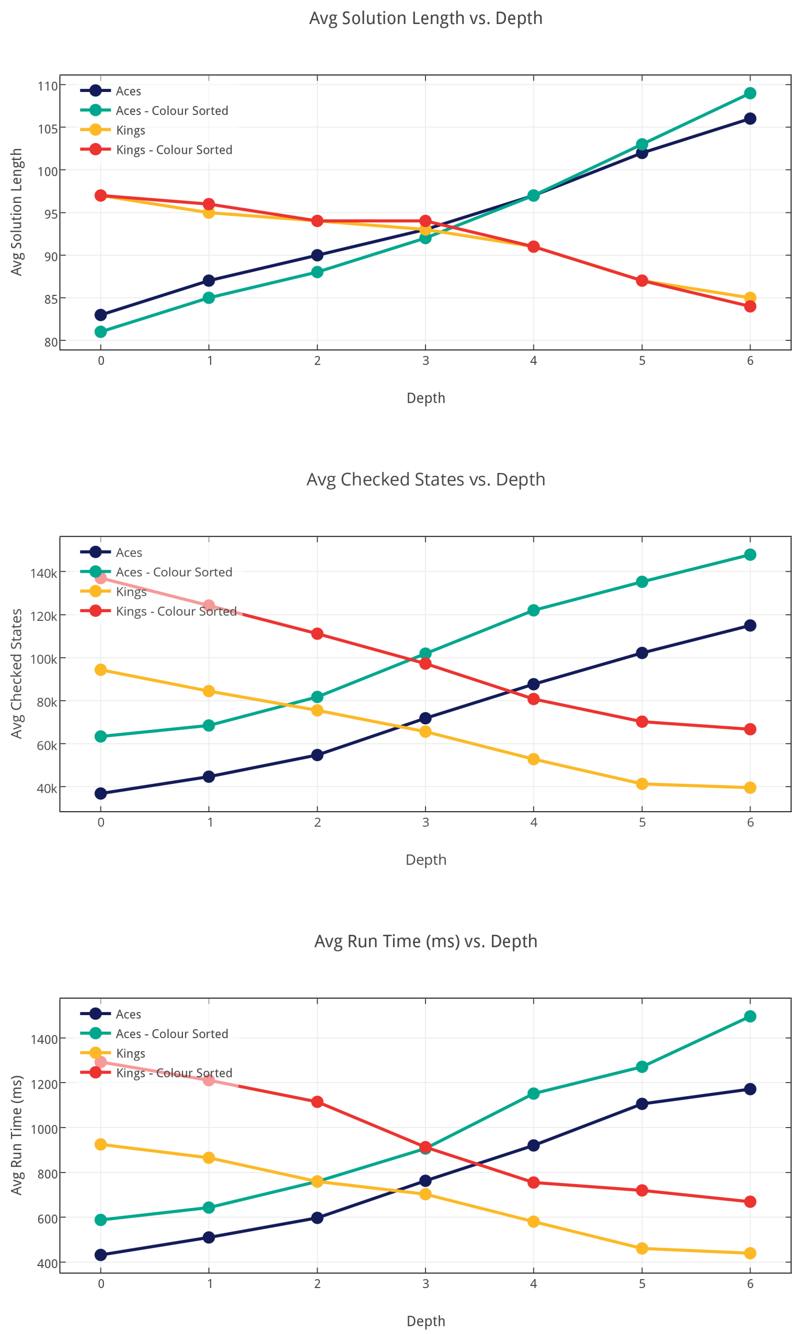

Figure 9: The effect of the placements of the aces and kings and the colour sorting characteristic on the three hardness indicators. Every plot is generated based on 14000 games, 2000 for each depth.

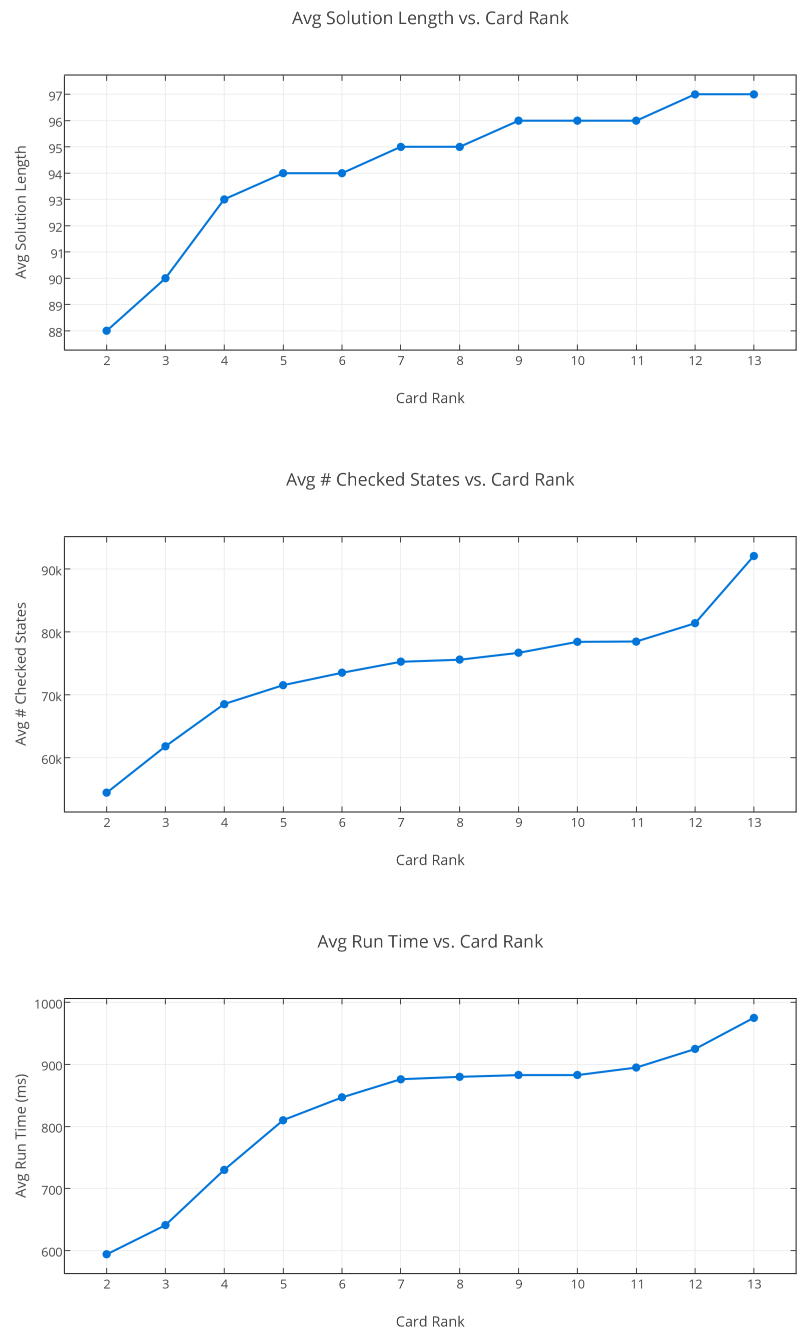

Figure 10: The effect of the blocking card characteristic on the three hardness indicators. The plot is based on 24000 games, 2000 for each card rank. The card ranks 11, 12 and 13 are equivalent to the card ranks jack, queen and king.

As it was previously mentioned in section 3.2.1, to determine which of the derived game characteristics potentially affect the hardness, a baseline for the three hardness indicators is necessary.

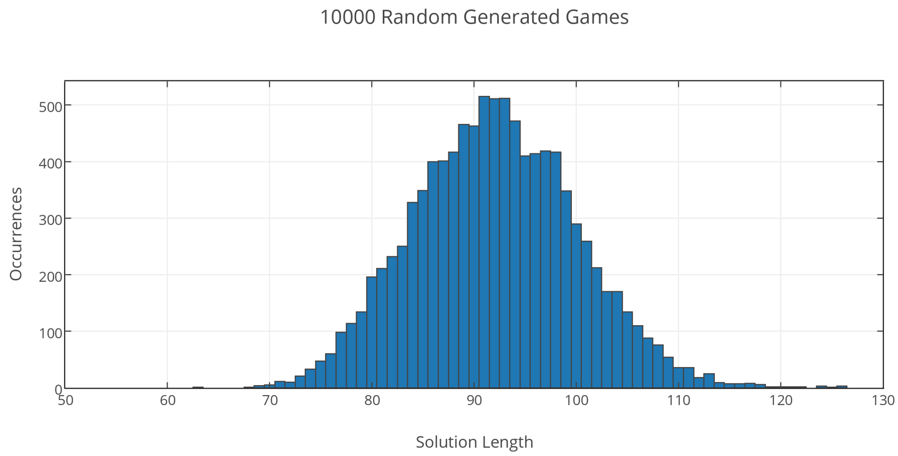

Figure 11: The number of occurrences sorted by solution length of 10000 random generated games. Avg solution length (SL) = 92; Median = 92; Mode = 91. See table 2 for more values.

Figure 11 displays the solution length of 10000 random generated games against the number of occurrences. The relevant values regarding the solution length could easily be calculated with this data. The same applies to the average number of checked states and the average run time in milliseconds for these 10000 games.

Figure 12 demonstrates the differences of the solution length of different types of games. The games that were generated with aces at depth 0 and the kings at depth 5 or 6, should generally be easier to solve, since figure 9 indicated that aces located at lower depths and kings at higher depths have shorter solution lengths. Combining these properties should make the games remain easy. Conversely, when the placement of the aces and kings are reversed (i.e. aces at depth 5 or 6 and kings at depth 0), the games should remain harder than average. This is likewise reflected in the first two histograms in figure 12, as the average solution length is higher. In addition, it seems that the harder the game, the higher the variation of the range of the solution length, since both the ranges of the second and last histogram is greater than the first histogram. Furthermore, the first two histograms have similar form, as both figures resembles a normal distribution. That is not the case for the colour sorted games, as it seems like that there are two peaks, which is fascinating.

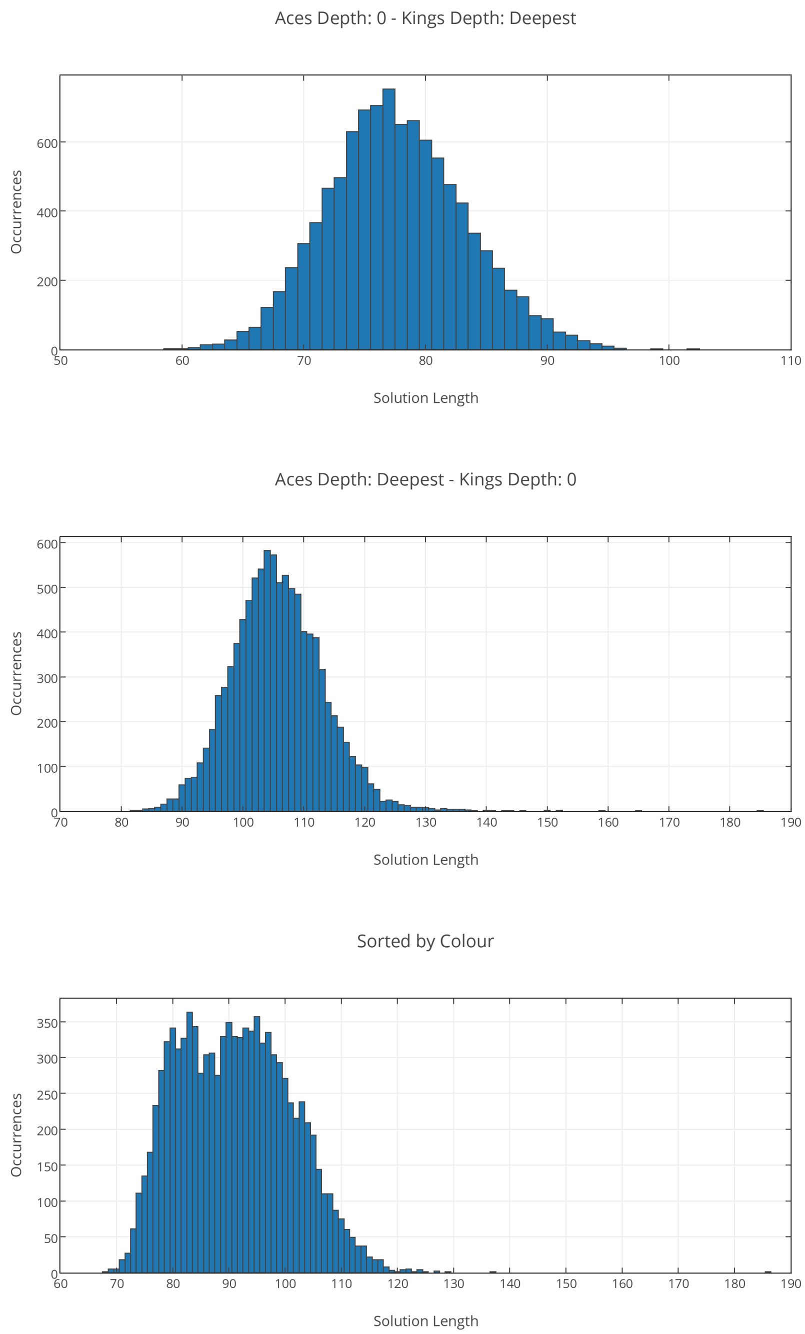

Figure 12: The number of occurrences vs. solution length of 10000 games for each type. First fig.: Avg SL = 77; Median = 77; Mode = 77; Second fig.: Avg SL = 106; Median = 105; Mode = 104; Last fig.: Avg SL = 91; Median = 91; Mode = 83

Table 2 displays some interesting values for the four game characteristics concerning the solution length (SL). The last column displays the p-value obtained from the t-test. This value measures the the significant differences between two data sets. If the p-value is equal to or smaller than the significance level (α, traditionally 0.05), it suggests that the differences of two data sets are significant.

Since the type 'Random' functions as the baseline, it would achieve a p-value of 1.0, as there are no differences between two of the same data sets.

The abbreviation of the game type 'A0KDeep' means that the four aces were placed at depth 0 and the four kings were placed at the lowest positions in the cascades. This is an ideal placement for the aces and kings (fig. 9), so lower values than the baseline are expected. The table proves that this seems the case, as every value is indeed lower than the baseline values. Furthermore, the T-test also achieves a p-value of zero, which means that the differences are significant.

The abbreviation 'ADeepK0' is the exact opposite of the game type 'A0KDeep', thus the placements of the aces and kings are reversed, which is a problematic placement of the aces and kings (fig. 9). The table indicates that every value is higher than the baseline, which was expected. As a result, the p-value is close to zero with a value of 2.934E-256, which implies that there is a significant difference.

The values of the colour sorting characteristic are less significantly different, as the average solution length is closed to the baseline value. Nonetheless, the p-value is still lower than the significance level with a p-value of 6.031E-14, implying that the colour sorting characteristic produced values significantly different from the baseline values.

With regard to the blocking cards characteristic, table 2 reveals that the values between the baseline and the blocking cards characteristic differentiate significantly, due to a p-values close to zero. However, it is important to note that the blocking cards characteristic is less significant, compared with the placements of the aces and kings characteristic, as the p-values are higher. This was likewise reflected in the plots in figure 10.

Type	Min SL	Max SL	Avg SL	SL T-test (p-value)
Random (Baseline)	63	126	92	1.000
A0KDeep	59	102	77	0.000
ADeepK0	82	277	106	2.934E-256
Colour sorted	68	186	91	6.031E-14
Blocking Cards 2	65	132	88	1.929E-31
Blocking Cards K	75	131	97	7.288E-49

Table 2: Relevant values for several types of games regarding the solution length (SL). Type 'Random' functions as the baseline. The types 'A0KDeep' and 'ADeepK0' are based on the placements of the aces and kings characteristics, 'Colour sorted' is based on the colour sorting characteristic and 'Blocking Cards 2, K' is based on the blocking cards characteristic. 10000 games were generated per type.

Table 3 displays some interesting values for the four game characteristics concerning the number of checked states (CS). The table displays similar patterns as in table 2, as easier games have lower values than the baseline and harder games have higher values than the baseline.

Recall, that figure 9 indicated that the colour sorting characteristic had a much greater effect on the number of checked states and the run time, compared with the solution length. This is likewise reflected in table 3, as the p-value of the colour sorted characteristic is lower, compared to the p-value of the colour sorted characteristic in table 2. Nevertheless, the results in table 3 indicate that all four characteristics result in a significant difference with the baseline values, since the p-values are close to zero.

Type	Min CS	Max CS	Avg CS	CS T-test (p-value)
Random (Baseline)	3974	218505	65641	1.000
A0KDeep	1927	185854	21454	4.5116E-189
ADeepK0	26118	805686	126336	1.239E-229
Colour sorted	3638	229868	100516	7.123E-78
Block Cards 2	3426	218531	54459	8.523E-15
Block Cards K	13944	207701	92077	8.484E-77

Table 3: Relevant values for several types of games regarding the number of checked states (CS). Type ‘Random’ functions as the baseline. The types ‘A0KDeep’ and ‘ADeepK0’ are based on the placements of the aces and kings characteristics, ‘Colour sorted’ is based on the colour sorting characteristic and ‘Blocking Cards 2, K’ is based on the blocking cards characteristic. 10000 games were generated per type.

Table 4 displays relevant values regarding the number of checked states. Once again, identical patterns emerges with the previous 2 tables, as low p-values are produced, which signals that the compared data sets were significantly different from the baseline.

Type	Min RT (ms)	Max RT (ms)	Avg RT (ms)	RT T-test (p-value)
Random (Baseline)	89	2697	727	1.000
A0KBot	62	1555	272	2.0977E-195
ABotK0	275	8313	1203	2.0758E-220
Color sorted	97	3401	958	6.0857E-53
Block Cards 2	87	1895	594	2.484E-16
Block Cards K	180	2362	975	1.866E-51

Table 4: Relevant values for several types of games regarding the run time (RT). Type 'Random' functions as the baseline. The types 'A0KDeep' and 'ADeepK0' are based on the placements of the aces and kings characteristics, 'Colour sorted' is based on the colour sorting characteristic and 'Blocking Cards 2, K' is based on the blocking cards characteristic. 10000 games were generated per type.

5 Evaluation

This section describes how the predictive model was developed and evaluates how well the performance was of the model.

5.1 Allocation of hardness scores to games

Section 4 indicated that all derived characteristics affect the hardness of FreeCell games, when considering the solution length, the number of checked states and the run time of the solver. As a result, hardness scores could be allocated to each game, based on the characteristics. The hardness scores could function as a model that could a priori determine the hardness of FreeCell games, without attempting solving the game first.

The hardness score is defined by the placements of the aces and kings, the blocking cards and the number of incorrect ordered cards. The incorrect ordered cards were taken into account, as the order of the cards play a crucial role for the hardness of FreeCell games, given how the game is played. The colour sorting characteristic is not included in the model, since it was difficult to convert this characteristic into a hardness score.

5.1.1 The aces and kings

Since lower values for the hardness indicators were achieved, when the aces are placed higher in the cascades (i.e. lower depth) and the kings lower in the cascades (i.e. higher depth), it would be possible to convert this feature into a score. When aces are placed on top of the cascades, meaning that they are exposed (depth is equal to 0), it is the most favourable position for the aces, as they could be moved directly to the foundations. This results in zero penalties. How many cards the aces are away from being exposed could be considered as the penalty score. In other words, the amount of cards on top of an ace, which is essentially the same as the depth variable is considered as the penalty score for the concerning ace. The exact opposite penalty was applied to the kings. Kings placed at the lowest positions in the cascades are given no penalties, as that is an ideal position for the king, whereas exposed kings will be penalised with the number of cards being blocked by the kings (penalty of five or six, since the cascades have a length of six and seven cards). An example game is given below to illustrate this score allocation based on the placements of the aces and kings (fig. 13).

Observing the placements of the aces, it can be seen that the `AH` is placed at depth zero, `AS` at depth five and both `AC` and `AD` at depth three. This lead to a score of 11 for the aces. As for the kings, `KC` blocks six cards, `KS` blocks one card, `KH` blocks four cards and `KD` blocks zero cards, which results also in a score of 11. So the total score acquired, based on the aces and kings is equal to 22.

```
5C 9S 4S 5S 7D JC 6D KD
TC AS 9C KS 7H 2D QC 4C
6H 7S 8C TS 6C 8H 4H 8S
TH JD AC AD TD 2H 8D 2C
QH 5D 3C QD 4D 5H 9D KH
3S 3D 3H 7C 9H JH QS JS
6S AH 2S KC
```

Figure 13: An example game to illustrate the hardness score allocation. The hardness score allocated by the model is equal to 112.

5.1.2 The blocking cards

Section 4 illustrated that higher values for the three hardness indicators were achieved, the higher the card ranks are blocking the aces. Therefore, the conversion of this characteristic to a score has been decided as the card *ranks* blocking the aces. Observing the example game in figure 13, it can be seen that `AH` is not blocked by other cards, since it is exposed, `AS` is blocked by the `7S`, `AC` is blocked by the `3C` and `AD` is blocked by the `QD`. Summing all ranks of the blocking cards, results in a score of 22.

5.1.3 The order of the cards

Given the game rules and the winning condition of FreeCell, it is logical that the order of the cards is crucial. The game should be easier to play, when more cards are placed correctly and in order. Hence, the number of disordered cards is also taken into account. So the score is based on the number disordered cards in each cascade.

So starting with the first cascade, it can be seen that there are no cards placed in order, meaning that there are seven disordered cards. The same applies for the second cascade. The third cascade has the `9C` followed by the `8C` and the `3H` followed by the `2C`, meaning that there is a total of four cards placed in order, thus there are only three disordered cards. Subsequently, no ordered cards are seen in the fourth cascade, resulting in seven disordered cards. The fifth cascade has four disordered cards, since the `7H` is followed by the `6C`. Lastly, all remaining cascades likewise have seven disordered cards. As a result, the total disordered cards and thus the score is equal to 46.

5.1.4 The model

Revising the allocated scores on the previous sections, it could be noticed that the score obtained with the aces and kings is equal to the blocking cards score. A maximum score of 48 could be achieved, when only examining the aces and kings, while a maximum score of 52 could be achieved for the blocking cards (four aces blocked by four kings). This seems a bit unreasonable, since the placements of the aces and kings have a greater impact on the hardness than the blocking cards according to the results. So to make the score ratios more fair, it has been decided to *double*[1] the scores obtained by the aces and kings.

Therefore, the developed model would allocate the following hardness score to the game in figure 13:
The total score for the aces and kings calculated in section 5.1.1 is equal to 22. As it has been decided that this score will be doubled, the total score is thus equal to 44. Section 5.1.2 achieved a total score of 22 and the score obtained by the order of the cards in section 5.1.3 is equal to 46. Hence, the hardness score for the given game is equal to 112.

5.2 Evaluation of the developed model

The model should be evaluated to investigate, whether it delivers reasonable results. Recall that the three hardness indicators were the solution length produced by the solver, the number of checked states and the run time in milliseconds. These three indicators could function as the hardness measure generated by the solver for each game. So, the performance of the model could be evaluated by calculating the correlation strengths between the hardness score allocated by the model with the three hardness indicators.

The correlation strength was determined by the 'Pearson's Correlation Coefficient *r*'. This coefficient measures how well two variables correlates to one another. The ranges of the 'r' value are between -1 and 0 for negative correlations, and between 0 and 1 for positive correlations. The strength of the correlation could be described with three levels: r-values with values ranging from 0.7 to 1 are considered strong, values ranging from 0.4 to 0.7 are considered moderate and values below 0.4 are considered as weak.

Lets assume that the hardness score has a perfect linear relation with the solution length. That would result to a scatter plot where a linear line could be fitted perfectly, meaning that the score has a perfect correlation ($r = 1$) with the solution length and thus most likely likewise with the hardness of FreeCell games, as the solution length was considered as one of the hardness indicators. In other words, the performance of the derived model depends on the dispersion or the so called *strength* of the data points. The stronger the strength, the higher the r-value and the better a line could be fitted to the data, which most likely implies that the score correlates strong with the solution length.

100000 random games were generated for the evaluation. Figure 14 displays the correlation strengh between the solution length produced by the solver and the hardness score allocated by the model. The *direction* of the data seems to be positive, as the solution length increases, when the hardness score also increases. The *form* of the data seems to be linear, as the data follows a straight line pattern. The *strength* of the correlation between the solution length and the hardness score

[1] the scores could be multiplied with other numbers, but multiplying by two has been chosen. See section 7 for more information

seems to be moderate, as the r-value is equal to 0.4393. Thus, the derived model seems to have a positive correlation with the solution length produced by the solver with moderate strength.

Still, of the 100000 data points, three outliers can be seen in the figure. A possible reason for these outliers is that there is no guarantee that the solver finds optimal solutions for the given games. It only yields solution lengths lower than the average, as mentioned in section 3.3.1. It might be the case, that for those three games, unfortunate values were found by the solver, which cause these outliers.

Fitting a linear line results in the formula: $f(x) = 0.219x + 66.9$
The *Root Mean Squared Error* (RMSE) is the average vertical distance of a data point from the fitted line. This variable has the same unit as the quantity plotted on the y-axis. A RMSE value of 2.660, thus implies that the average error of the fitted line is three moves.

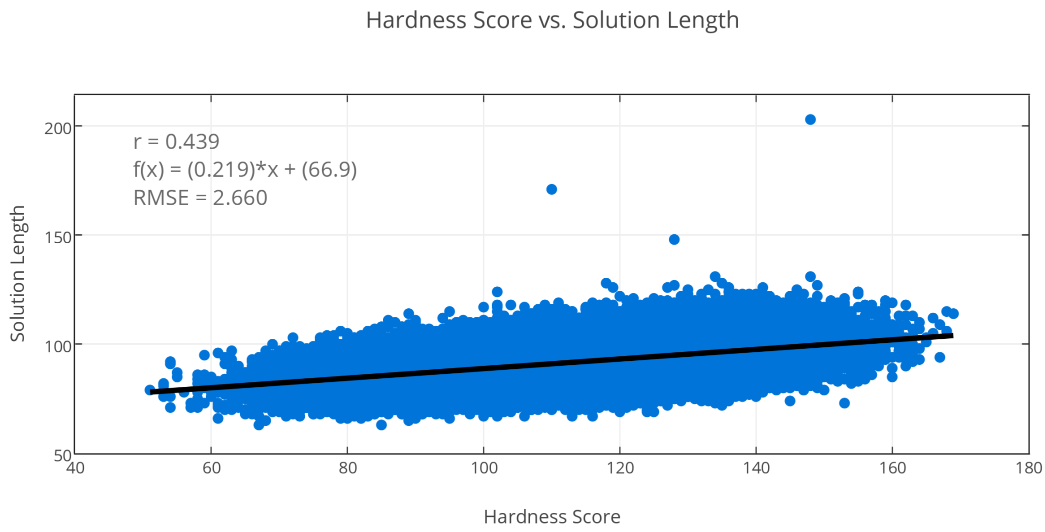

Figure 14: The correlation strength between the solution length produced by the Shlomifish FreeCell solver and the hardness score allocated by the developed model based on 100000 random games. The correlation strength seems to be moderate, as the r-value is equal to 0.439.

Figure 15 displays the correlation strength between the number of checked states and the hardness score allocated by the model. Similar features could be detected with the previous figure. The direction of the data is positive, as the number of checked states increases, when the hardness score also increases. The form of the data is linear, as the data follows a straight line pattern. However, the dispersion of the data points is greater, than in figure 14, which implies that the strength of the correlation is weaker, with a r-value equal to 0.366. In other words, the derived model seems to have a positive correlation with the number of checked states. However, the r-values implies that the correlation between these two variables are on the weak side.

The best fit of the data points with a formula of $f(x) = 838x - 30000$ achieves a RMSE of 189.737, implying that the fitted line has an average error of 190 checked states, compared with the data points. This amount seems low, which is a positive sign, given the tremendous range wherein the number of checked states are covered.

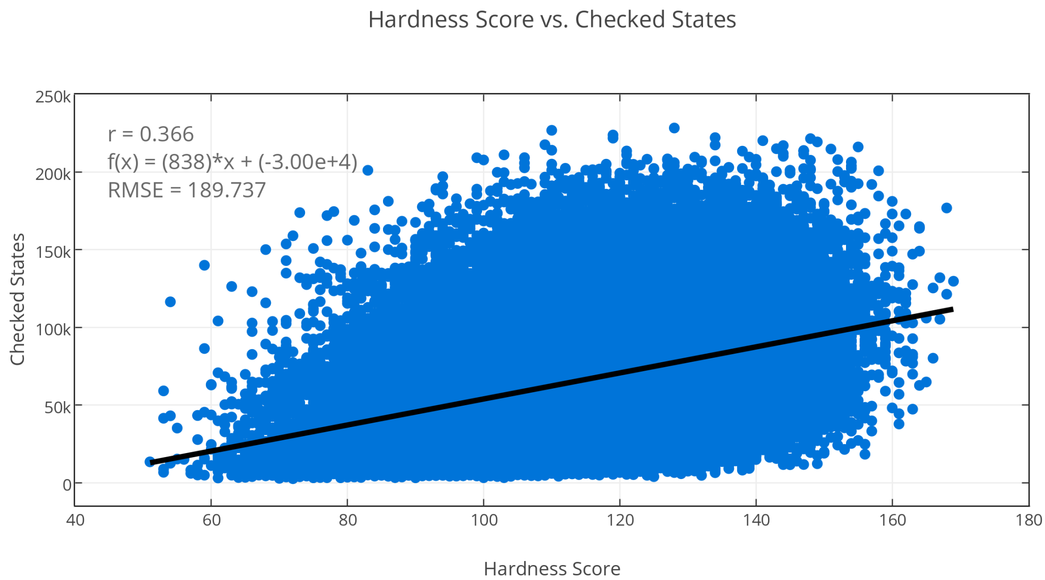

Figure 15: The correlation strength between the the number of checked states produced by the Shlomifish FreeCell solver and the hardness score allocated by the developed model based on 100000 random games. The correlation strength seems to be weak, as the r-value is equal to 0.366.

Figure 16 seems familiar, since it possesses resembling features, like the previous two figures. The figure displays the strength of the correlation between the run time of the solver and the hardness score. Once again, the data points establishes a positive and linear pattern, since the fitted line is positive and linear. The r-value is equal to 0.366, which implies as well that the correlation strength between the two variables is weak.

The best fit for the data points, described with the formula $f(x) = 7.96x - 213$, achieves a RMSE value of 17.869. This value is negligible considering the fact that this value is equivalent to approximately 18 ms.

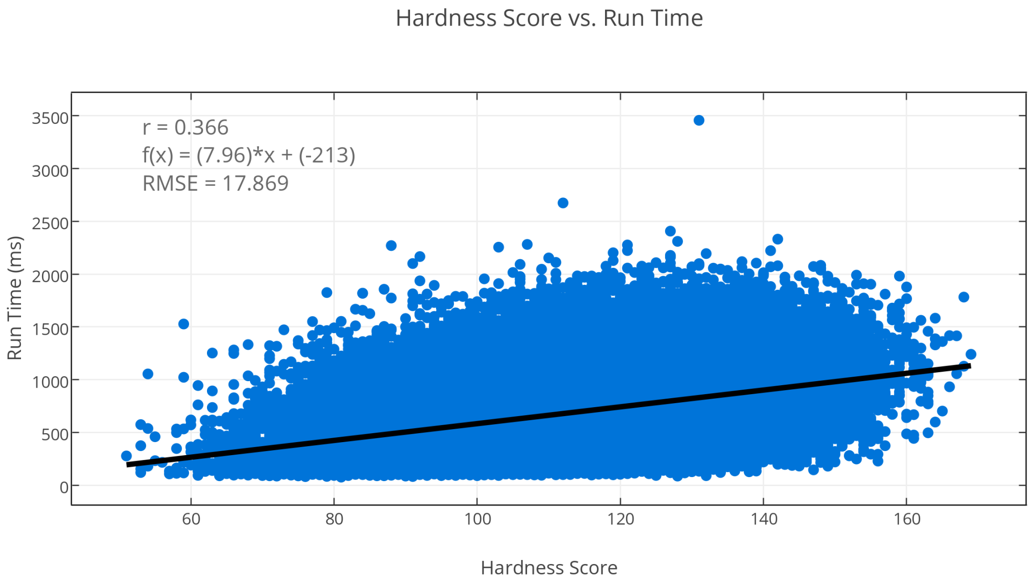

Figure 16: The correlation strength between the run time of the Shlomifish FreeCell solver and the hardness score allocated by the developed model based on 100000 random games. The correlation strength seems to be weak, as the r-value is equal to 0.366.

6 Conclusion

This thesis presented an attempt to predict the hardness of FreeCell games in terms of its characteristics. The research question 'Which game characteristics affect the hardness of FreeCell games and what type of FreeCell games are hard to solve?' functioned as an aid to develop a model that could allocate hardness scores to games. During this research, a total of four different game characteristics were investigated. The examined characteristics consist of the placements of the aces, the placements of the kings, the blocking cards and the colour sorting characteristic. The hardness of each game was determined by the solution length produced by the solver, the number of checked states and the run time in milliseconds.

The results suggest that the placements of the aces and the kings have the greatest impact on the three hardness indicators, as their p-value were relatively lower and the variation/ranges of the values of the indicators were likewise greater. Additionally, the colour sorting characteristic seems to have a greater effect on the number of checked states and the run time of the solver, than the solution length. Nevertheless, it could be concluded that all four investigated game characteristics affected the solution length, the number of checked states and the run time of the solver, since all values were significantly different from the baseline (p-values were close to zero and thus less than the threshold value α). Therefore, it is most likely the case that all four characteristics affect the hardness of FreeCell games.

As a result, a model that could allocate hardness scores based on the derived characteristics by observing the initial state of the game was developed. The purpose of this model was to indicate the hardness of FreeCell games a priori. The model achieved the highest correlation with the solution length indicator, with a r-value of 0.439, followed by a r-value of 0.366 for both the number of checked states and the run time indicators. This suggests that the correlation strength between the three hardness indicators and the hardness scores allocated by the model was weak to moderate. Therefore, it seems that the developed model may not be a good hardness indicator for FreeCell games.

7 Discussion

This section describes the flaws and issues, that were encountered during this research and how this potentially could be prevented. Moreover, some possible improvements are mentioned. Lastly, some future work directions are presented.

7.1 Flaws and issues

As the overall progress throughout this research led to decent results, some areas could have some improvements. Firstly, not all functions of the Shlomifish FreeCell solver was examined and utilised, as there are lots of parameters available that could be adjusted. It may result in more optimal results, when solving games with other parameter settings. Secondly, the program that examined the uniqueness of games could be improved. At the moment, the program is only capable of informing games, that are the *exact* same game. So games are not unique, when two games have all cards placed at the exact same positions, including colours and suits. However, games could in principle still be the 'same', as in that the same solution length, the number of checked states, the run time and the hardness score are produced. The occurs when only the colours and/or the suits are different. An example is given below, to clarify this (fig. 17).

```
QC 7D 7S 4C 3D 2C 3H KS      QS 7H 7C 4S 3H 2S 3D KC
TD TH JC 8S TC 6S 8H AD      TH TD JS 8C TS 6C 8D AH
TS KC 7H QD 6C 9D 8D 9C      TC KS 7D QH 6S 9H 8H 9S
KH 3S QH 6H 2H 4H 9H QS      KD 3C QD 6D 2D 4D 9D QC
4D 5S KD 3C 8C 7C 2D AC      4H 5C KH 3S 8S 7S 2H AS
9S 6D JD 4S 2S 5D 5H JS      9C 6H JH 4C 2C 5H 5D JC
AS AH 5C JH                  AC AD 5S JD
```

Figure 17: Two example games that would produce the same values for the three hardness indicators and the hardness score allocated by the model, while they are not the same games.

The two example games above are not exactly the same, as the clubs and the spades are reversed and the hearts with the diamonds. Therefore, the current 'uniqueness' detector would not inform that these games are the same. However, if both games were solved and analysed, then all produced values would be exact the same, since FreeCell is based on colours and not on suits. Hence the 'uniqueness' detector should be improved to handle all cases. Thirdly, the game generator should be extended, so that it is possible to generate games based on a given hardness score as input. In this manner, the predictive model could be examined more precisely, leading to more reliable conclusions, which could improve the performance of the model. Lastly, recall that in section 5.1.4 the scores obtained from the aces and kings were adjusted, so that the ratios were more fair. Unfortunately, the chosen value of two was baseless. It is even possible that the adjustment made the predictive capability of the model worse. Better was to execute multiple test runs with different values, so that the best value could be used for the model.

7.2 Future Research

It is recommended that further research should be undertaken in the several areas. Firstly, this research has focused mainly on the aces and the kings. The remaining card ranks also need to be investigated in order to have a better understanding of the effect, every card rank has on the hardness. As a result, a more accurate model would be developed, as more cards will be incorporated to determine the hardness score.

Secondly, besides the three hardness indicators that was used in this research, a fourth indicator could be taken into account, namely the minimal number of free cells that is necessary to solve a certain game. It might be the case that if more free cells are required to solve the game, then it implies that the game was harder to solve. To support this assumption, more research regarding the number of free cells is essential.

Subsequently, more research should be done regarding the developed model, as most of the allocations of the scores based on the characteristics were counted linearly. It is possible that linear counting is for some properties not appropriate. So more research should be done to adjust the scores.

Furthermore, as the evaluated model produced some outliers, for instance in figure 14, more research should be done to examine these outliers, since these data points are the most interesting.

Finally, further research should be done in people playing FreeCell. It would be beneficial, if the model and the three hardness indicators could eventually be linked to different levels of hardness, determined by people (human hardness indicator). Lets assume that there are three human hardness levels: hard, medium and easy. It is still unclear, which ranges of values for the solution length, the number of checked states and the run time correspond with the hard, medium and easy levels. So relationships between the hardness indicators with a 'human hardness indicator' is desirable.

References

Amit Benbassat, Achiya Elyasaf, and Moshe Sipper. More or less? two approaches to evolving game-playing strategies. In *Genetic Programming Theory and Practice X*, pages 171–185. Springer, 2013.

Ivan Bratko. *Prolog Programming for Artificial Intelligence*. Pearson Education Limited, 2012. ISBN 978-0-321-41746-6.

Chris Chan. Helping human play freecell.

Achiya Elyasaf, Ami Hauptman, and Moshe Sipper. Evolutionary design of freecell solvers. *Computational Intelligence and AI in Games, IEEE Transactions on*, 4(4):270–281, 2012.

Shlomi Fish. Shlomi fish's homepage, 2015a. URL `http://www.shlomifish.org/`.

Shlomi Fish. Freecell solver, 2015b. URL `http://fc-solve.shlomifish.org/`.

George Heineman. http://broadcast.oreilly.com/2009/01/january-column-graph-algorithm.html.

George Heineman, Gary Pollice, and Stanley Selkow. *Algorithms in a Nutshell*. O'Reilly Media, Inc., 2008. ISBN 059651624X, 9780596516246.

Malte Helmert. Complexity results for standard benchmark domains in planning. *Artificial Intelligence*, 143(2):219–262, 2003.

Holly Jimison, Misha Pavel, James McKanna, and Jesse Pavel. Unobtrusive monitoring of computer interactions to detect cognitive status in elders. *Information Technology in Biomedicine, IEEE Transactions on*, 8(3):248–252, 2004.

Holly B Jimison, Misha Pavel, and Jesse Pavel. Adaptive interfaces for home health. In *Proceedings of the International Workshop on Ubiquitous Computing for Pervasive Healthcare*, 2003.

Angela McFarlane, Anne Sparrowhawk, and Ysanne Heald. *Report on the educational use of games*. TEEM (Teachers evaluating educational multimedia), Cambridge, 2002.

ShlomiFish. personal correspondence.

H Wang et al. Rating logic puzzle difficulty automatically in a human perspective. In *Proceedings of DiGRA Nordic 2012 Conference: Local and Global–Games in Culture and Society*, 2012.