Algorithmic Analyses of Card-Discarding Type Games

Hironori KIYA

Supervisor

Professor Hirotaka ONO

2021 A Doctoral Course
in
Department of Mathematical
Informatics
Graduate School of Informatics
Nagoya University

January 2021

Contents

Preface

With the advent of the AI era, informatics research on recreational games has attract wide attention. Indeed, it is reported that AI programs for several recreational games such as chess, Shogi and Go succeeded in beating even top professional human players. It is a remarkable success of informatics in the sense that the victories are based on several fundamental information/computer technologies such as machine learning, simulations and hardware developments. In other words, the treatment of "games" in such research might however be rather a benchmark problem for the achievement of AI technology or performance evaluation than the research object; games themselves are not studied as the central objects of research but used as the measures of performance evaluation. Apart from such a treatment, "games" themselves have long attracted the interest of many researchers. A typical research field about "games" is game theory, which was launched as a tool for modeling economic activities. However, due to the original purpose of the development of game theory, recreational games are unfortunately not the main subject of research in game theory. As above, the study of entertainment games is often treated as a secondary research topic.

Combinatorial game theory is a branch of mathematics that studies winner and loser in games themselves. However, most of the studies in combinatorial game have been conducted on games with abstract game which we are not very familiar with, and on difficult games even if all program in the world cannot compute the winner. Is there a game that is neither too computationally hard nor too abstract? I consider that possible candidates are so-called card-discarding games.

This thesis is devoted to analyses of the time complexity of winner decisions in card-discarding games, which are widely played in the world, using the each player's hand as input. First, I consider TANHINMIN, which is a simplified version of the widely played game DAIHINMIN in Japan. I present a linear-time algorithm that determines which player has a winning strategy after all cards are distributed to the players in TANHINMIN game and its variant. Furthermore, this thesis is also concerned with how I obtain a winner for the imperfect information variant of TANHINMIN game. I newly introduce an oracle model in which the oracle provides partial information about the opponent's hand. Interestingly, when players can get partial information of the opponents' hands via oracle, the winning player can find a winning strategy as if it is the (perfect information) TANHINMIN. Furthermore, I show various results about other relationships between the power of oracles and the existence of a computable winning strategy. I also considered multiplayer card-discarding games. I newly introduce open-hand BABANUKI based on the ordinary BABANUKI. I consider the winning strategy of open-hand variant BABANUKI. Although the 2-players case is almost obvious, the 3-players case is not, and I give a necessary and sufficient of the existence of the winning strategy. Furthermore,

for 4-players case, there is a configuration where an endless-loop phenomenon, so-called "repetition draw", occurs. Finally, I consider winning strategy 2-players SHICHINARABE game. Through the graphical generalization, I show that an winner decision algorithm of the general game can be computed in linear time, and show that the time complexity of the game under some local rules.

Hironori Kiya

Acknowledgments

This thesis would not have been possible without the help of many people, whom I would like to acknowledge here.

First of all, I would like to express my sincere appreciation to Professor Hirotaka Ono of Nagoya University for his supervising this thesis. Without his suggestion, continual encouragement and his careful supervision, this thesis would not have been completed. For his greatly help including advice and encouragement, I gain the many opportunities to present my work outside of the university, which gave indispensable comment and advice for this thesis.

I would also like to thank Professor Mutsunori Yagiura of Nagoya university, Professor Taro Takimoto of Kyushu university, Professor Yota Otachi of Nagoya university Professor Tesshu Hanaka in Chuo University, Professor Hiroshi Eto of Kyushu university, Professor Yannan Hu of Tokyo university of science, Mr. Katsuki Ohto, Dr. Koki Suetsugu of National Institute of Informatics, Mr. Hama Vitor Mitsuo Fukushigue of Nagoya university, Ms. Hang Dong of Nagoya university, Mr. Genki Sakoda of Nagoya university, Mr. Hisato Yamanaka, Mr. Masahiro Okubo, Ms. Naoka Fukuzono and Mr. Yasuhide Maei for useful discussions and kindly encouragement.

I would also like to thank the all members in Professor Ono's Laboratory, Professor Yagiura's Laboratory and Professor Otachi's Laboratory for their hospitality during this 3 years. I feel grateful for having the great opportunity to be shared a lot of idea and knowledge during every seminar, research meeting, and in every conversation.

Finally, but not less important, I wish to express my great thank to my family for their heartfelt cooperation and encouragement.

Chapter 1

Introduction

1.1 Background

With the advent of the AI era, "games," especially recreational games, have gained significance in several aspects. For example, a typical application of AI in games is to design game-playing algorithms based on machine learning. Indeed, some AI-based game-playing algorithms are superior to the top human players in many abstract games [52,53], demonstrating the significant development of AI technologies. Although these approaches seem to be new, game research has a long history. One traditional game research is game theory, which mathematically models and analyzes the behavior of human decision-making in social life as a "game." Game theory is a fundamental area of economics research, and was used by the winners of Nobel Prizes in Economics in 1994 and 2012 [2]. However, these are not studies of recreational games themselves. The former study only treats games as a benchmark for machine learning, taking advantage of the good match between machine learning reward systems and games. In other words, most of the interest is in the performance of the machine learning method, not the game itself. The main purpose of the latter research is to facilitate the analysis of social phenomena through games. In contrast, recreational games themselves are not completely outside of game theory; however, they are almost entirely outside the scope of interest of game theorists.

In contrast to these research fields, combinatorial game theory is a discipline that evaluates the mathematical properties of games themselves. Combinatorial game theory is an academic system that originated from Bouton's work on Nim and developed greatly with the Sprague-Grundy theorem in the 1930s. Combinatorial game theory mainly characterizes mathematically the "winning way" for abstract games. In this thesis, similar to combinatorial game theory, we analyze the game itself. Our goal is to analyze the combinatorial games and similar games using an algorithmic approach called winner decision.

1.1.1 Combinatorial game

Combinatorial games are defined as *two-player games* with *sequential game*, *perfect information*, and *no chance* [51].

Two-player games: The games played by precisely two players. Almost all the two-player board games or two-player card games we play are *zero-sum games* in which each participant's gain or loss of utility is exactly balanced by the losses or gains of the utility of the other participants. In zero sum games, if one player is the winner, the other is the loser.

Sequential game: In these games, one player chooses their action before the others choose theirs. The latter players must have some information about the first player's choice, otherwise the difference in time would have no strategic effect.

Perfect information game: These are games played without hidden information. Each player, when making any decision, is perfectly informed of all the events that have previously occurred to the other player(s), including the "initialization event" of the game.

No chance game: These games do not include chance elements such dice rolls or coin flips.

Many recreational games we play as abstract games can be classified as combinatorial games. For example, Shogi and Go are combinatorial games.

1.1.2 Winner decision

Winner decision is an algorithmic approach to combinatorial games. The winner decision is described algorithmically as follows:

GAME 1 Winner decision of combinatorial game

Input: Game Situation

Output: The player who has the winning strategy (if there is no winner, none)

In other words, winner decision refers to an algorithm that outputs a player who has a winning strategy for a game in Nash equilibrium (which is given as the input). In combinatorial games, if the game is finite, it always has a winning strategy of either the first or the second player [50]. Thus, the answer to the question "is there an algorithm that can gain a winner decision" is always "yes, there is," if we do not have to worry about execution time or execution area. Although there exists an algorithm to determine the winner, it does not necessarily mean that it is simple and tractable.

Therefore, the main concern in the winner decision is the efficiency of the algorithm, that is, whether the algorithm can be executed in a shorter calculation time or less calculation space as in the theoretical computer science. For example, in case of algorithms that output correct answers for the same input, an algorithm with a shorter calculation time and less calculation space can be said to be a more "excellent" algorithm. The measure of this calculation time is called *time complexity*, and similarly, the measure of this calculation space is called *space complexity*. Time complexity is typically evaluated by counting the number of elementary operations performed by the winner decision algorithm, supposing that each elementary operation takes a fixed amount of time to perform. As the running time of a winner decision algorithm may vary among different inputs of the same size, we commonly considers the worst-case time complexity, which is the maximum amount of time required for inputs of a given size. We commonly consider this complexity using "order notation," for example, $\mathcal{O}(n)$, $\mathcal{O}(2^n)$, and so on. where n is the input size in units of bits needed to represent the input. The time complexities of a winner decision algorithm are classified according to the function appearing in the order notation. For example, an algorithm with multi-computation steps that is less than or equal to a constant multiple of the size of the input expressed by $\mathcal{O}(n)$ in order notation is called a *linear-time* algorithm.

Let us return to our original goal; what is the time complexity of the winner decision for a combinatorial game? The answer is, of course, or unfortunately, it depends on the game.

For example, well-known Nim and its variants have simple ways to compute in polynomial time their nimbers, which enable the determination of the winning player [12]. In fact, a kind of nim-type games are one of the largest class of two-player games that have polynomial-time winner decision algorithms; the winner decision of other games, including "Moore game" and "Chomp" for which linear time algorithms are known [9,22,37]. However, this does not mean that the winner decisions of almost all combinatorial games are easy for a winner decision.

Another example, "Go," is also a famous combinatorial game. The time complexity of winner decisions for "Go" is EXPTIME-hard in which there is no polynomial-time algorithm for the winner decision [44]. In fact, although there exist winner decision algorithms for generalized versions of many popular board games such as chess and Shogi, such algorithms hardly run in polynomial time, because they are also known to be EXPTIME-complete to determine the winner [44]. There is a large gap between the existence of winner decision algorithms and the existence of polynomial-time winner decision algorithms.

From such a perspective, many researchers study the computational complexity of the winner decision problems for generalized versions of popular recreational games, but unfortunately, most of the results, including the above examples, are shown to be intractable.

These examples show that even within the same combinatorial game, there is a large gap in the calculation time for the winner decision. What causes this large gap? To examine this large gap, we now present the two games rule: "Nim" and "go"

For Nim and Nim-type games are played with a pile of stones, where the players take turns removing stones from the pile, and the player who removes the last stone wins. In the "Go" game, one player uses white stones and the other, black. The rule is as follows: The players take turns placing the stones on the vacant intersections of a board. stones of opponent player's can be removed from the board if the group of stones are surrounded. The winner is determined by counting each player's territory at the game end. Now we carefully consider the large gap about time complexity; Nim, where the input is a set of the number of stones per pile, is relatively easy to compute, while Go, where the input is the board situation, is relatively hard to compute. Actually, many other studies on combinatorial games have highlighted this trend. For example, Kayles, which is Nim-type game, can be done in polynomial time using the Sprague-Grundy theorem [24], while "Node Kayles," which is also a nim-type game with board, is PSPACE-hard to decide the winner [11, 16, 48].

Now we are ready to consider one of our goal, what happens to the computational complexity for other inputs? It is not known. Most of the interest in computational complexity is limited to games for which the input is a set of element or board information in combinatorial games, and there are very few studies on other games, including those we usually play. This is our main motivation for this research. We examine the time complexity of a widely played card-discarding game.

1.2 Card-discarding game

Card-discarding games, which are also called shedding-type card game, are games in which a player's objective is to empty their hand of all cards before any other player does.

In this thesis, we mainly consider combinatorial card-discarding games. Some card-discarding games are categorized as combinatorial games, that is, two-player perfect information games; however most of them are not. For example, UNO is a multi-player game and imperfect information game. In this thesis, we define combinatorial game variants for popular card-discarding games and consider its time complexity for winner decisions.

The three card-discarding games that we address in this thesis are the TANHINMIN game, the BABANUKI game with open-hand, and the SHICHINARABE game. In the next subsection, we will briefly introduce each of these games.

1.2.1 TANHINMIN

TANHINMIN, which means Single Pauper, is a card-based combinatorial game, which was proposed by Nishino in order to investigate the mathematical properties of DAIHINMIN [40]; DAIHINMIN (which means Grand Pauper), or DAIFUGO (which means Grand Millionaire), is a popular playing-card game in Japan. The basic rule of DAIHINMIN is quite simple, and many similar games are played all over the world. For example, it is similar to the Chinese game Dou Dizhu, Big Two and Zheng Shangyou, to the Vietnamese game Tien Len, and to Western card games like President, also known as Capitalism and Asshole, and The Great Dalmuti [1,3]. Not only that, it has attracted attention in the table of AI for Games. In fact, the DAIHINMIN programming competition is held at the University of Electro-Communications in JAPAN every year. Although DAIHINMIN AI programs are getting stronger every year [41, 42], the mathematical nature of the game itself is still mostly unknown. DAIHINMIN games contain various special rule that make it exciting but also difficult to analyze. For this reason, the TANHINMIN, which is one of the simplest variant of DAIHINMIN, was introduced for DAIHINMIN research.

DAIHINMIN is a card discarding game. The basic rule of DAIHINMIN is as follows: at the beginning of the game, all cards are distributed to the players. A player starts the game by discarding any set of cards with a same number (e.g., one 4, two 4's or three 4's, if the player has three 4's), and each player discards one or more cards in turn according to the strength system of cards, or skips the turn. A player can discard only a set of cards when it is stronger than the set of cards that are discarded by the previous player. If no player discards, then the turn ends and the player who last discarded a set of cards can start a new turn by discarding a set of cards as above. After several turns, the first player that has discarded all his/her cards is the winner. The basic rule of TANHINMIN is the same as DAIHINMIN, but it is simplified in the following two senses. (1) A player can discard not more than one cards but a single card, and (2) the strength system of the cards is just a total order based on the face values.

1.2.2 BABANUKI

BABANUKI is a popular card-based game in Japan. The basic rule of BABANUKI is quite simple, and many similar games are played all over the world [1]. For example, "Old maid", a card game played in Britain, is similar to BABANUKI.

BABANUKI and Old Maid are card discarding games. The basic rule of BABANUKI is as follows: We have a set of 2-card pairs (25-26 pairs if we play with regular cards) and one Old maid card (JOKER in this game). At the beginning of the game, all the cards are distributed to the players where they are faced down. Until the end of the game, the cards of each player are hidden to the other players. After all the cards are distributed, each player can discard pairs of cards anytime if he/she has them in his/her own hands. If the first player has a same card with the drawn card, then he/she discards the drawn card together with the same card. Otherwise, the player adds the drawn card into his/her hand. In any case, this finishes the first player's turn and the next (second) player's turn starts; the second player drawn a card from the first player's hands, and so on. Eventually, the last player draws a card from the previous player's hands and ends his turn. Then the first player again draws a card from the last player's hands. After several turns, the player that has first discarded all his/her cards is the winner.

1.2.3 SHICHINARABE

SHICHINARABE is a popular multiplayer imperfect information game played with playing cards in Japan, and similar games such as Fan-Tan and Sevens are widely played in other countries. The general rule of SHICHINARABE is as follows.

We use a deck of cards, that is, the 52 cards with JOKER in total. We can give some special role to JOKER, though we omit the explanation here. The players have a common goal, which is to put the cards of each suit in sequential number order like $(\spadesuit A, \spadesuit 2, \ldots, \spadesuit 10, \spadesuit J, \spadesuit Q, \spadesuit K)$, which is called a *layout*. All the cards are (almost) evenly dealt to the players. Each player removes all sevens from his/her hand and put them to the centers of the layouts as setting up. A layout is made by the players' putting cards down one by one next to the seven in sequential order; a player can put a card next to a card already put in a layout in his/her turn. A player who cannot or does not want to place a card selects a "pass." When a player has run out of passes (normally on their fourth pass), they place all their remaining cards in their places on the playing space. The player that is first to lose their whole hand wins.

1.3 Related work

In this section, we introduce related studies that we could not fully explain in the previous sections.

1.3.1 Combinatorial game

In the previous sections, combinatorial games were divided by what input is; combinatorial games are generally divided into *impartial game* and *partisan game* by what player can do. Impartial games are the games such as Nim, Sprouts, Kayles, Quarto, Cram or Chomp, which distinguish only between first player and second player; partisan(or partizan) games are the games such as Go, chess or Shogi, which distinguish not only between first player and second player but also players state. [9, 51],

Impartial games can be analyzed using the Sprague–Grundy theorem, stating that every impartial game under the normal play convention is equivalent to a nimber [23, 54]. The representation of this nimber can change from game to game, but every possible state of any variation of an impartial game board should be able to have some nimber value. If the nimber of a game is known, it is easy to determine the victory of the game. Therefore, research to determine the nimber of a game has been actively conducted as a substantial victory determination [4, 24]. However, to compute nimber of a game is not always easy. For example, Node Kayles is PSPACE-complete to compute nimber [11, 26].

In combinatorial game theory, a game is partisan (sometimes partizan) if it is not impartial game but combinatorial game. That is, some moves are available to one player and not to the other. Most combinatorial games that we play as a recreational game are partisan game for example Go, Shogi, Chess is partisan. Generally speaking, partisan games are more difficult to analyze than impartial games, as the Sprague–Grundy theorem does not apply. In fact, many games have been known to be difficult. For example, Shogi, Chess, Go, and Cheaker are known to be EXPTIME-complete, and Othello, Amazon, Konane, Cross Purposes and many partisan games are known to be PSPACE-complete [20, 21, 25, 27, 30, 31, 45, 46]. On the other hand, there are some games which can be solved only in small size or in limited situations. For example, Shogi and chess and xiangqi are EXPTIME-complete as described above, but it is known that it is possible to make a winning decision in a constant time in a probabilistically limited situation by using an oracle [29]. There is also an attempt to strongly solve the game using search for the actual game size. These studies have been conducted since the 1990s, when computers became more accurate. For example, checker, nine men's morris, Go-Moku, Qubic, and quixo have been strongly solved [6–8, 43, 49, 57]. However, this is only in very limited circumstances, and analysis is hopeless for general partisan games. To calculate the winner partisan game like an impartial game, we often use a value called "game value." The game value is a very useful indicator in a "last player to move wins (loses)" type of partisan game [51]. Like a nimber, if the game value of a game is known, it is easy to determine the victory of the game [15, 18, 39].

In that sense, a card-discarding game is a partisan game, since each player's hand is not always common (it may contain common elements, but it doesn't share them). We do not write about the details of the combination game here any more. If you want to study combinatorial game theory, see [5], [9] or [51].

If you are looking for more detailed research, you may want to look at Fraenkel's collection of combinatorial game bibliographies [19].

1.3.2 Non-combinatorial games

In this section, we focus on non-combinatorial game. First of all, we see single-player games as simpler games than combinatorial games.

Recall that it always has a winning strategy of either the first or the second player in combinatorial games without draw game if the game is finite.

If the game is not a combinatorial game, the situation changes; for example, if there are three or more players, there is a case that no one has a winning strategy. This is due in large part to the fact that your opponent's strategy will change depending on whether the other players are behaving cooperatively or not. Therefore, researchers study multiplayer impartial game by adding some assumption to determine the unique game results: defined rank system, alliance matrix or preferences among the players [32–36, 55].

The same situation is occurred in imperfect information game. In other words, the cases that no one has a winning strategy also exist in imperfect information game. This is because players do not know what their opponent has done in the past or what they might do in the future, so players do not know what their best strategy is. Therefore, for such imperfect information games, there are analyses that assigns probabilities to each game that is treated as a perfect information game [10, 13].

No sequential game, named simultaneous game, is a similar situation to imperfect information game; Since players do not know what their opponent has done simultaneously, so players do not know what their best strategy is. Therefore, assigning probabilities to each opponent player's strategy has been used like in imperfect information games. Simultaneous games are of central interest in (economical) game theory, because many economic activities does not occur alternately. See A and B for a detailed explanation of simultaneous turn games. See [38] and [47] for a detailed explanation of simultaneous games.

Chance game such as SUGOROKU and Backgammon is known to be analyzed by expectation. For example, poker is a chance game, but this poker is said that by using a strategy based on expected value, the player has acquired the ability to surpass the top level human players [14, 56].

1.4 Contribution

We show a winner decision algorithm for a Card-discarding game, on which there is few study before. First, we present results on TANHINMIN, the basic hand-consuming entertainment game, in Chapter 2, and its variants in Chapter 3. Then, we analyze Oracle TANHINMIN as a new approach for incomplete information entertainment games in Chapter 4. Then, we tackle BABANUKI as an approach to multiplayer entertainment games in Chapter 5. Finally, in Chapter 6, we show results on SHICHINARABE, a game is not only card-discarding games nut also using board. Further details are given below.

At first, we present an efficient algorithm that determines which player has a winning strategy of the 2-player TANHINMIN and its variant in Chapter 2 and Chapter 3. It computes a kind of game status defined by the maximum matching sizes of configuration graphs in linear time from sorted hands. As seen in the previous section, most of the existing results about the winner decision for natural non-impartial (partisan) games are hardness ones, and most of positive results (polynomial-solvability) are restricted to impartial games, as far as the authors know. In this sense, polynomial-solvability of the winner decision of 2-player TANHINMIN, a natural non-impartial game, could be a rare and interesting example of the computational complexity of game analyses. Furthermore the result could be practically useful for designing a strong DAIHINMIN AI program, for example. Although TANHINMIN is a restricted variant of DAIHINMIN, it still follows the basic framework of the rules and the restriction does not spoil all the mathematical natures of the game. Actually, when a configuration of DAIHINMIN satisfies (1) every player has no duplicated cards in his/her hand, (2) every player has no special card in his/her hand, the game can be considered a configuration of TANHINMIN. This implies that the algorithm could be implemented to DAIHINMIN AI programs and the result is also a first-step to design a linear-time winner decision algorithm for a 2-player generalized DAIHINMIN.

Since these variant of TANHINMIN is a 2-player perfect information game without draw, either the first or the second player always has a winning strategy, which means that the winner decision is possible. On the other hand, DAIHINMIN, which is the original game of TANHINMIN, is an imperfect information game; there does not necessarily exist a player having a winning strategy. This is a motivation to investigate an imperfect variant of TANHINMIN in Chapter4.4. As mentioned in the previous section, probability-based analysis is a common analysis method for incomplete information game analysis. However, this analysis method has two problems; one is that that it cannot be used when the mother set is unknown, that is, when there is no information about the input of the other player. The other is that it is rare to output a definitive solution. In Chapter4.4, we model TANHINMIN with structural oracles to identify the essential information to construct a winning strategy. As will be described later, in TANHINMIN without any information, no player has a winning strategy like many imperfect information games. We think that the setting “no information” is quite rare in real game playing situations; players can get some information of their opponents’ hands, e.g., how many cards she has, whether she has a specific card, and so on. The oracle model that we propose in this thesis can qualify and quantify the information that each player can receive during plays. Under the proposed oracle models, we obtain several results. Interestingly, when players can get partial information of the opponents’ hands via oracle, the winning player can find a winning strategy as if it is the (perfect information) TANHINMIN. The idea of the proof is based on a detailed analysis of the winner decision algorithm of the perfect information TANHINMIN in Chapter 2. Furthermore, we show various results about other relationships between the power of oracles and the existence of a computable winning strategy, which is shown in Fig. 4.1.

Next, we try to analyze multi-player game with perfect information. As mentioned in the previous section, for analysis of multiplayer games, it is common to use an analysis that assumes a preference relationship among other players. However, the preference relations among other players do not necessarily exist, much less observe them correctly from other player. In this study, we investigated the behavior of a partizan game that focuses only on "maximizing one’s own rank," as is commonly used in game theory. We will observe a multiplayer game on the subject of BABANUKI which is easy to compute winner decision in two players case. It is played in the manner that players play BABANUKI with cards faced up. This makes the game a perfect information game, and it becomes worth considering optimal strategies; we consider the winning strategy of open-hand variant BABANUKI. Although the 2-players case is almost obvious, the 3-players case is not, and we give a necessary and sufficient condition of the existence of the winning strategy. Furthermore, for 4-players case, there is a configuration where an endless-loop phenomenon, so-called “repetition draw”, occurs.

Finally, we introduce 2-players SHICHINARABE game with perfect information. As stated above, SHICHINARABE is a multiplayer imperfect information game in which players try to achieve one of the two victory conditions: play all cards, or all other players run outs. For the former only, research on UNO [17] and for the latter only, analysis based on game value can be cited as precedents, but there are no examples of research on games with both of these victory conditions. In addition, this game is a card-discarding game in which the player who discards their hand wins, but it is also a game in which the players refer to the information on the board. While research has been conducted on card-discarding games and board games, there are little research has been conducted on combined games.

In this study, We define SHICHINARABE value in Chapter 6, which is an extension of game value. This value covers not only games in which the player who loses a move loses, but also games in which the player who loses all his cards wins. Using this SHICHINARABE value, we have shown that ordinary SHICHINARABE can be solved in linear time. Furthermore we model a graphical generalization of SHICHINARABE and investigate the time complexity of the winner decision. Through the graphical generalization, we present a linear-time algorithm that can decide the winner of a given ordinary SHICHINARABE instance; the ordinary SHICHINARABE is shown to be an easy game. On the other hand, SHICHINARABE has many variations on how to use JOKER cards and with/without tunnel rules.

We formally define such rules in the graphical models, and we see the effect of the strength of rule sets. Concretely, we pick up trees and planar graphs as typical layouts of a board, and investigate the time complexity for natural combinations of graph classes and rule sets. As a result, the winner decision of graphical SHICHINARABE on trees is proved to be solvable in polynomial time, whereas that on planar graphs is shown to be hard to solve; the winner decision is NP-hard in general, and it is even PSPACE-hard if we adopt a generalized tunnel rule.

Chapter 2

TANHINMIN and its variant

2.1 Introduction

DAIHINMIN, which means Grand Pauper, is a popular playing-card game in Japan. TANHINMIN is a simplified variant of DAIHINMIN, which was proposed by Nishino in 2007 in order to investigate the mathematical properties of DAIHINMIN. In this chapter, we consider a 2-player generalized TANHINMIN, where the deck size is arbitrary n. We present a $\mathcal{O}(n)$ time algorithm that determines which player has a winning strategy after all cards are distributed to the players.

2.2 TANHINMIN Rules and Notations

2.2.1 The rule of TANHINMIN

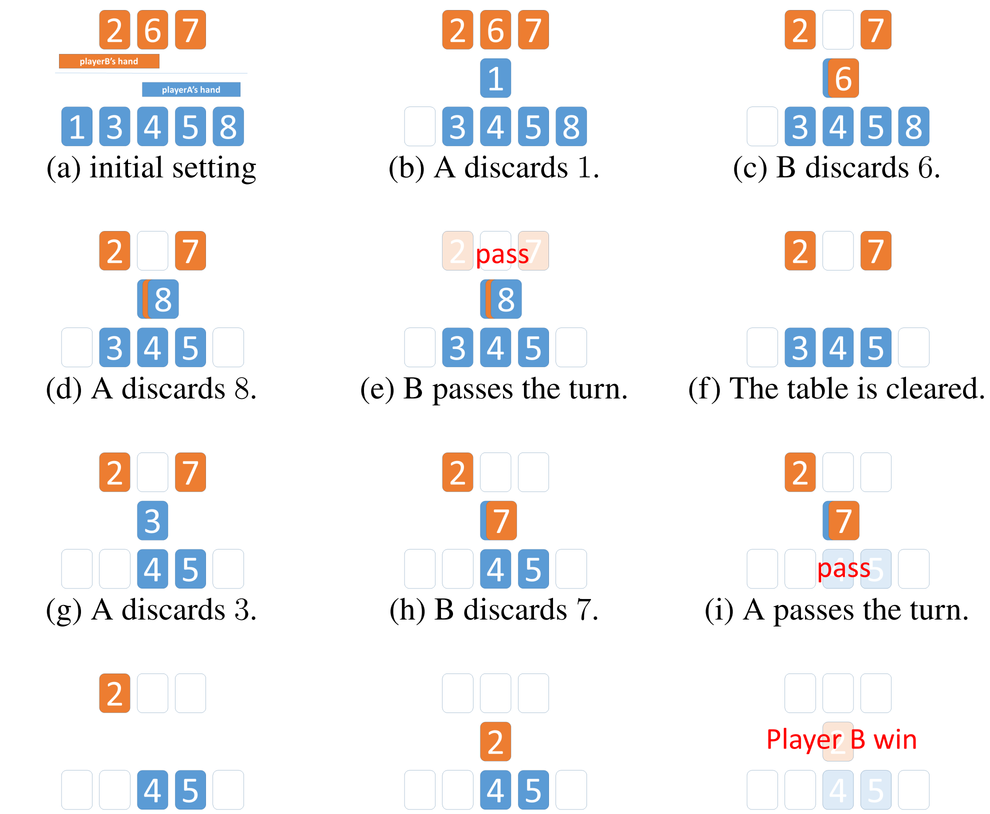
We first model a game of TANHINMIN. Let $[n] = \{1, 2, \ldots, n\}$ be the set of card faces, where the number represents its strength. Card 1 is the weakest, and 2 is stronger than 1 but weaker than 3, and so on. TANHINMIN use cards with the strength relationship. As we see later, a player can discard a stronger card than the card at the table. In the game of TANHINMIN, the faces of some cards can be same, but in the following, to simplify the explanation, we assume that no two cards have a same face; a set of cards is not a multiset but just a set. Note that this assumption does not change the nature of TANHINMIN. We just distinguish two cards of "3", as "3_1" and "3_2", for example. This assumption does not change the nature of TANHINMIN. In fact, even if we have two or more cards of a number ("3", for example) , all the proofs in this paper work by ordering these cards as 3_1, 3_2, $\ldots$. The rule of basic TANHINMIN game that we consider in this paper is as follows: All the cards are distributed to players. At the beginning of the game, there is no card on the table (empty). Active player in his/hers turn may discard a card in hand onto the table. The player to discard a card is called *active*, and the other is called *non-active*. Once the active player discards a card, the turn ends. Then the active player becomes non-active, and the non-active player becomes active, and the next turn starts. A card to discard must be stronger than the lastly discarded card on the table, which we call a *table card*. If the table is empty, then any card can be discarded. If the player of the turn does not have a card to discard or does not want to discard any card, he/she selects "pass". Then let the table be empty and go to the next turn. The player that first discards all the cards in his/her hand is the winner.

Note that our rule allows an active player to select pass even when the table is empty. After the player does so, the table remains empty and the turn moves to the opponent player, which may cause infinite consecutive passes, though it seems to be useless. In fact, when we consider winning strategies, we can assume that two or more consecutive passes do not occur; since the players' hands do not change after two consecutive passes, if a player has a winning strategy with the existence of consecutive passes, he/she has a winning strategy without consecutive passes. That is, we assume that the players avoid consecutive passes in the following.

In Figure 2.1, we show a play example of 2-player TANHINMIN. Here we explain the detail as follows: At first, cards $1, 3, 4, 5, 8$ are distributed to player A (blue player), $2, 6, 7$ are distributed to player B (orange player) (Fig. 1 (a)). We suppose that player A is the first player, and the table is empty; any card can be discarded. Thus player A has five options: discarding card 1, 3, 4, 5 or 8. In this example, player A discards card 1 (Fig. 1 (b)), and the turn moves to player B. Next, player B can discard stronger cards than 1 at the table; player B has four options: discarding card 2, 6, 7 or passing the turn. In this example, player B discards card 6 (Fig. 1 (c)), and the turn moves to player A. Then player A has two options: discarding card 8, or passing the turn. In this example, player A discards card 8 (Fig. 1 (d)). Then player B has only one option: passing the turn (Fig. 1 (e)). Since player B here selects pass, the cards on the table are cleared (Fig. 1 (f)), and player A plays next. In this setting, player A chooses to discard card 3 (Fig. 1 (g)), and so on. The game continues to Fig. 1 (k), where player B discards the last card 2; since player B first finishes discarding all his/hers card, player B is the winner (Fig. 1 (l)).

2.2.2 Graph Model of TANHINMIN

We assume basic knowledge of graph theory. Let $G = (V, E)$ be a graph, where V is the set of vertices and E is the set of edges. All the graphs that we consider in this paper are *bipartite*, that is, there is a bipartition (V_0, V_1) of V such that $E \subseteq \{(p, q) \mid p \in V_0, q \in V_1\}$. To specify the bipartition, we write $G = (V_0, V_1, E)$ instead of $G = (V, E)$. For graph G and a vertex v of G, $N_G(v)$ denotes the set



(a) initial setting (b) A discards 1. (c) B discards 6.

(d) A discards 8. (e) B passes the turn. (f) The table is cleared.

(g) A discards 3. (h) B discards 7. (i) A passes the turn.

(j) The table is cleared. (k) B discards 2. (l) B wins the game.

Figure 2.1: A play example of 2-player TANHINMIN

of neighboring vertices to v in G, that is, $N_G(v) = \{u \in V \mid \{u, v\} \in E\}$. We sometimes use notation $N(v)$ instead of $N_G(v)$ if the graph that we consider is clear. For $S \subseteq V$, $N_G(S)$ similarly denotes the set of vertices neighboring to any vertex in S of G, that is, $N_G(S) = \bigcup_{v \in S} N_G(v)$. We similarly use $N(S)$ instead of $N_G(S)$. For graph $G = (V, E)$ and $v \in V$, let $G \setminus v$ denote a graph obtained by deleting v and its incident edges. For a graph $G = (V, E)$, a subset M of E is called *matching* if no two edges in M share an end. For a graph G, let $\mu(G)$ denote the size of a maximum matching.

We first fix a turn to consider. In the turn, we suppose that P_0 is the active player and P_1 is the non-active player. At the turn, X_0 and X_1 respectively denote the cards belonging to P_0 and P_1, and by r the top card on the table. These provide sufficient information to describe the situation of the turn; triplet (X_0, X_1, r) define the configuration of the turn.

Note that in a play of TANHINMIN cards on table are sometimes cleared, and then r is empty. In such a case, we virtually consider that 0 is at the top of the cards on table. For example, in Figure 1, $X_0 = \{1, 3, 4, 5, 8\}$, $X_1 = \{2, 6, 7\}$ and $r = 0$ at (a), and $X_0 = \{2, 6, 7\}$, $X_1 = \{3, 4, 5, 8\}$ and $r = 1$ right after (b).

We then give a graph model of TANHINMIN; for a configuration, we construct several bipartite graphs. The vertices correspond to cards in $X_0 \cup X_1 \cup \{r\}$, and use the same symbols to represent them. Here, we introduce a general way to define a bipartite graph whose vertices are cards: For two disjoint pair of card sets V_0 and V_1, we can define a bipartite graph according to the strength of the cards by $G = (V_0, V_1, E(V_0, V_1))$ where $E(V_0, V_1) = \{(v_0, v_1) \mid v_0 \in V_0, v_1 \in V_1, v_0 > v_1\}$. Throughout the paper, all the bipartite graphs are constructed in this way. Since an ordered bipartition (V_0, V_1) of vertices can determine the structure of $G = (V_0, V_1, E(V_0, V_1))$, we just write $G = (V_0, V_1)$ instead of $G = (V_0, V_1, E(V_0, V_1))$ below. For configuration (X_0, X_1, r), we then construct bipartite graphs $G_0 = (X_0, X_1)$ and $G_0(r) = (X_0, X_1 \cup \{r\})$. Similarly, we define $G_1 = (X_1, X_0)$ and $G_1(r) = (X_1, X_0 \cup \{r\})$. Here, graph $G_0(r)$ represents which cards P_0 can discard for cards in $X_1 \cup \{r\}$. Graph G_1 represents which cards P_1 can discard for cards in X_0. If $X_0 = \emptyset$ or $X_1 = \emptyset$, P_0 or P_1 is obviously the winning player, respectively. Thus we assume both X_0 and X_1 are nonempty in the following analyses.

As we see later in Lemma 2.2, the winner of TANHINMIN is determined by the maximum matching sizes of two graphs obtained from $G_0(r)$ and G_1, that is, $\mu(G_0(r) \setminus \min X_1)$ and $\mu(G_1 \setminus \min X_0)$, where $\min X_0$ (resp., $\min X_1$) denotes the weakest card of X_0 (resp., X_1). Since these graphs play important roles in the winner decision, we name these the configuration graphs. More specifically, if P_0 (resp., P_1) is the active player and P_1 (resp., P_0) is the non-active player, we call $G_0(r) \setminus \min X_1$ (resp., $G_1(r) \setminus \min X_0$) the configuration graph of the active player, and call $G_1 \setminus \min X_0$ (resp., $G_0 \setminus \min X_1$) the configuration graph of the non-active player.

2.3 Winner decision of TANHINMIN

Now we are ready to show how we can decide who has a winning strategy for a given configuration. More concretely, we prove the following.

Theorem 2.1. *Given a configuration of 2-Player TANHINMIN with n cards, we can decide which player has a winning strategy in $\mathcal{O}(n)$ time.*

This theorem holds by Lemmas 2.2 and 2.5.

Lemma 2.2. Given a configuration (X_0, X_1, r) of 2-player TANHINMIN, where P_0 is the active player and P_1 is the non-active player, P_0 has a winning strategy when $\mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0)$ holds, and P_1 has a winning strategy otherwise.

Inside of the proof of Lemma 2.2, we use the following two lemmas. The proofs are shown later.

Lemma 2.3. Consider the situation where P_0 is the active player and $\mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0)$ holds. Then, P_0 has a strategy (i.e., a sequence of P_0's actions of discarding a card or selecting pass) such that the configuration graphs of the resulting configuration (X'_0, X'_1, r') eventually satisfy $\mu(G_1(r') \setminus \min X'_0) \leq \mu(G_0 \setminus \min X'_1)$ with $|X'_0| < |X_0|$, where P_0 is non-active and P_1 is active.

Lemma 2.4. Consider the situation where P_0 is the active player and $\mu(G_0(r) \setminus \min X_1) \leq \mu(G_1 \setminus \min X_0)$ holds. Then, for any action of P_0, P_1 has a strategy (i.e., a sequence of P_1's actions of doing nothing, discarding a card or selecting pass) such that P_1 wins (i.e., discards the last card) or the configuration graphs of the resulting configuration (X'_0, X'_1, r) satisfy $\mu(G_1(r') \setminus \min X'_0) > \mu(G_0 \setminus \min X'_1)$ with $|X'_0| < |X_0|$, where P_0 is non-active and P_1 is active.

By using these lemmas, we prove Lemma 2.2 as follows. The proof is based on the induction on $|X_0|$.

Proof of Lemma 2.2. **(Base step)** We see the case $|X_0| = 1$ and let x_0 be the unique card (vertex) in X_0. We first consider the case where $\mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0)$ holds and P_0 is the active player. We will show that P_0 wins in this case. Since P_0 has only x_0 in hand, if x_0 is stronger than r, P_0 wins by discarding x_0. Thus we consider the other case: $x_0 < r$ (i.e., r is isolated in $G_0(r)$) and P_0 cannot discard x_0; P_0 passes the turn.

We see the situation right before P_0's pass. Since $|X_0| = 1$ implies $\mu(G_0(r) \setminus \min X_1)$ is at most 1, $1 = \mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0) = 0$ holds. Since $\mu(G_0(r) \setminus \min X_1) = 1$ with isolated r, x_0 can be matched with x'_1, which is the weakest in $X_1 \setminus \min X_1$. This implies that P_1 has cards $x'_0 (\equiv \min X_1)$ and x'_1 in hand. In order to win, P_1 needs to discard at least two cards x'_0 and x'_1 before P_0 discards x_0. However, once P_1 discards either x'_0 or x'_1, P_0 can then discard x_0 in the next turn; P_0 wins.

We next consider the case where $\mu(G_0(r) \setminus \min X_1) \leq \mu(G_1 \setminus \min X_0)$ and P_1 is the non-active player. We will show that in this case P_1 wins. By $|X_0| = 1$, $G_1 \setminus \min X_0 (= G_1 \setminus X_0)$ is a graph with no edge, and we have $0 = \mu(G_1 \setminus \min X_0) \geq \mu(G_0(r) \setminus \min X_1) = 0$, which implies that $G_0(r)$ has no edge between x_0 and any vertex in $X_1 \cup \{r\}$ except $\min X_1$. This means that P_0 must pass this turn, and P_1 becomes the active player whose cards except $\min X_1$ are not weaker than x_0, with empty table. Then if P_1 has one or more cards other than $\min X_1$, he/she can discard all of them in the following turn. In fact, if P_1 plays such a card, which is placed on the table, the unique card x_0 of P_0 is not stronger than it; P_0 must select pass and P_1 keeps to be active, and we can apply the same argument repeatedly. Then, eventually the active player P_1 only has the last card $\min X_1$ with empty table. By just discarding it, P_1 wins.

By these, the statement of Lemma 2.2 holds for the base step $|X_0| = 1$.

(Induction step)

Assume that for $|X_0| \leq k$ the statement of Lemma 2.2 is true, i.e., the active player P_0 has a winning strategy when $\mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0)$ holds, and the non-active player P_1 has a winning strategy when $\mu(G_0(r) \setminus \min X_1) \leq \mu(G_1 \setminus \min X_0)$ holds.

Let $|X_0| = k+1$. If $\mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0)$, Lemma 2.3 indicates that the active player P_0 can play such that the resulting configuration (X'_0, X'_1, r') satisfies $\mu(G_1(r') \setminus \min X'_0) \leq \mu(G_0 \setminus \min X'_1)$ holds, where P_0 is non-active and $|X'_0| \leq k$. Then by Lemma 2.4, whatever active player P_1 plays, non-active player P_0 can play so that P_0 wins or the resulting configuration (X''_0, X''_1, r'') satisfies $\mu(G_0(r'') \setminus \min X''_1) > \mu(G_1 \setminus \min X''_0)$, where $|X''_0| \leq |X'_0| \leq k$, which is the winning condition of the non-active player P_0 by the assumption of the induction. Similarly, if $\mu(G_0(r) \setminus \min X_1) \leq \mu(G_1 \setminus \min X_0)$, by applying Lemmas 2.4 and 2.3 in this order, P_1 wins or the situation is reduced to the configuration (X''_0, X''_1, r'') satisfying $\mu(G_0(r'') \setminus \min X''_1) \leq \mu(G_1 \setminus \min X''_0)$, where P_0 is active and $|X''_0| \leq k$, which is the winning condition of the non-active player P_1 by the assumption of the induction.

By these, for $|X_0| = k + 1$ the statement of Lemma 2.2 is also true, which completes the proof. $\square$

Proof of Lemma 2.3. Suppose that P_0 is the active player and $\mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0)$ holds. Roughly there are two cases: P_0 has a stronger card than the table card r (i.e., (a) $N_{G_0(r)}(r) \neq \emptyset$), or not (i.e., (b) $N_{G_0(r)}(r) = \emptyset$).

For case (a), we focus on a maximum matching of $G_0(r) \setminus \min X_1$ where r is matched with a vertex in X_0. Such a maximum matching always exists by the property of maximum matchings. We then further divide the case into two cases: (a-1) there is a maximum matching of $G_0(r) \setminus \min X_1$ such that r is matched with a card $x \neq \min X_0$, (a-2) in every maximum matching of $G_0(r) \setminus \min X_1$, r is matched with $\min X_0$.

Case (a-1): In this case, let P_0 discard x, and r be replaced with x. Then P_0 and P_1 become the non-active and active players, respectively. The configuration moves to $(X'_0, X'_1, r') = (X_0 \setminus \{x\}, X_1, x)$. For the non-active player P_0, the configuration graph is $G_0 \setminus \min X'_1 = (X'_0, X'_1 \setminus \min X'_1) = (X_0 \setminus \{x\}, X_1 \setminus \min X_1)$, which is the graph obtained from $G_0(r) \setminus \min X_1$ by removing matching edge (x, r) in a maximum matching of $G_0(r) \setminus \min X_1$. That is, $\mu(G_0 \setminus \min X'_1) = \mu(G_0(r) \setminus \min X_1) - 1$. For the active player P_1, the configuration graph forms $G_1(r') \setminus$

$\min X_0' = (X_1, X_0' \setminus \{x, \min X_0'\} \cup \{r'\})$. Since $r' = x$ and $\min X_0' = \min X_0$, the configuration graph is equivalent to $G_1 \setminus \min X_0$ for the original (X_0, X_1, r). Namely, $\mu(G_1(r') \setminus \min X_0') = \mu(G_1 \setminus \min X_0)$. These and the initial inequality yield the following inequality: $\mu(G_0 \setminus \min X_1') = \mu(G_0(r) \setminus \min X_1) - 1 \geq \mu(G_1 \setminus \min X_0) = \mu(G_1(r') \setminus \min X_0')$, with $|X_0'| = |X_0| - 1$, which is the one in the statement of Lemma 2.3.

Case (a-2): In this case, in every maximum matching of $G_0(r) \setminus \min X_1$, r is matched with $\min X_0$. Then all the vertices in X_0 are covered in such a maximum matching, otherwise there exists an unmatched card $x(\neq \min X_0)$ in a maximum matching, and we can have a maximum matching that does not contain $(\min X_0, r)$ by replacing $(\min X_0, r)$ with (x, r), which is a contradiction. Thus, $|X_0| = \mu(G_0(r) \setminus \min X_1)$ holds. In this case, let P_0 discard $\min X_0$, and the configuration moves to $(X_0', X_1', r') = (X_0 \setminus \min X_0, X_1, \min X_0)$. The configuration graph of non-active player P_0 for (X_0', X_1', r') is $G_0 \setminus \min X_1' = (X_0', X_1' \setminus \min X_1') = (X_0 \setminus \min X_0, X_1 \setminus \min X_1)$, which is the graph obtained by removing a matching edge $(\min X_0, r)$ from $G_0(r) \setminus \min X_1$. Thus $\mu(G_0 \setminus \min X_1')$ is at least $\mu(G_0(r) \setminus \min X_1) - 1 = |X_0| - 1$. The configuration graph of active player P_1 is $G_1(r) \setminus \min X_0' = (X_1', X_0' \cup \{r'\} \setminus \min X_0') = (X_1, X_0 \setminus \{r'\} \cup \{r'\} \setminus \min X_0') = (X_1, X_0 \setminus \min(X_0 \setminus \min X_0))$. Thus the maximum matching size is at most $|X_0 \setminus \min(X_0 \setminus \min X_0)| = |X_0| - 1$. These implies $\mu(G_0 \setminus \min X_1') \geq |X_0| - 1 \geq \mu(G_1(r) \setminus \min X_0')$.

We now consider the case (b) where P_0 has no car stronger than the table card r, i.e., $N_{G_0(r)}(r) = \emptyset$. Then what P_0 can do is to pass the turn, and P_1 becomes the active player. Then P_1 has two options: passing the turn, or discarding a card x. If P_1 passes the turn, the configuration changes from the original (X_0, X_1, r) to $(X_0', X_1', r') = (X_0, X_1, 0)$ with active player P_0; the configuration graph of P_1 is the same as $G_1 \setminus \min X_0$, though that of P_0 changes from $G_0(r) \setminus \min X_1$ to $G_0(0) \setminus \min X_1$, which is obtained from $G_0(r) \setminus \min X_1$ by adding edges $(x, 0)$ for every $x \in X_0$. Thus, $\mu(G_0(r') \setminus \min X_1') \geq \mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0) = \mu(G_1 \setminus \min X_0')$ holds, which is reduced to case (a) above. Thus, in the following, we consider the case P_1 discards a card x, which is always possible; r is replaced with x, and the configuration becomes $(X_0', X_1', r') = (X_0, X_1 \setminus \{x\}, x)$ with active P_0. We further divide the case into (b-1) $x = \min X_1$, (b-2) $x \neq \min X_1$.

Case (b-1): In this case, P_0 must have a stronger card in $X_0' = X_0$ than $x = \min X_1$. This is because otherwise any card in X_0 is not stronger card than any card in X_1, which implies $\mu(G_0(r) \setminus \min X_1) = 0$ in the original configuration; it contradicts the precondition $\mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0) \geq 0$. The configuration is $(X_0', X_1', r') = (X_0, X_1 \setminus \min X_1, \min X_1)$, and $\min X_1' = \min(X_1 \setminus \min X_1)$, which is denoted by $\min_2 X_1$. The configuration graph of active player P_0 changes from the original $G_0(r)\setminus\min X_1 = (X_0, X_1\setminus\min X_1\cup\{r\})$ to $G_0(r')\setminus\min X_1' = (X_0, X_1\setminus\min_2 X_1)$. Note that $G_0(r')\setminus\min X_1'$ has $\min X_1$, whereas $G_0(r)\setminus\min X_1$ has $\min_2 X_1$ and isolated r. By $\min X_1 < \min_2 X_1$, if $G_0(r)\setminus\min X_1$ has an edge between some $x' \in X_0'$ and $\min_2 X_1$, $G_0(r')\setminus\min X_1'$ has an edge between some $x' \in X_0$ and $\min X_1$. Thus, $\mu(G_0(r')\setminus\min X_1')$ is at least $\mu(G_0(r)\setminus\min X_1)$. On the other hand, the configuration graph of non-active player P_1 changes from $G_1\setminus\min X_0 = (X_1, X_0\setminus\min X_0)$ to $G_1\setminus\min X_0' = (X_1', X_0'\setminus\min X_0') = (X_1\setminus\min X_1, X_0\setminus\min X_0)$, which is a subgraph of $G_1\setminus\min X_0$. Thus $\mu(G_1\setminus\min X_0) \geq \mu(G_1\setminus\min X_0')$ holds. By these, $\mu(G_0(r')\setminus\min X_1') \geq \mu(G_1\setminus\min X_0')$ holds with $N_{G_0(r')} \neq \emptyset$ and $|X_0'| = |X_0| = k+1$, which is reduced to case (a).

Case (b-2): $x \neq \min X_1$ holds. The configuration is $(X_0', X_1', r') = (X_0, X_1\setminus\{x\}, x)$, and $\min X_1' = \min(X_1\setminus\{x\}) = \min X_1$ holds, because $x = \min X_1$. The configuration graph of active P_0 becomes $G_0(r')\setminus\min X_1' = G_0\setminus\min X_1 = G_0(r)\setminus\min X_1$, by the assumption of case (b). That is, $\mu(G_0(r')\setminus\min X_1') = \mu(G_0(r)\setminus\min X_1)$. On the other hand, $\mu(G_1\setminus\min X_0')$ is at most $\mu(G_1\setminus\min X_0)$, because $G_1\setminus\min X_0'$ is a subgraph of $G_1\setminus\min X_0$. Thus, $\mu(G_0(r')\setminus\min X_1') > \mu(G_1\setminus\min X_0')$ and $N_{G_0(r')}(r') = \emptyset$ hold with active player P_0; the situation is reduced to case (b) itself but a smaller $|X_1'|$. This means that (b-2) cannot occur infinitely, and case (b-1) is essential.

Overall, all the cases eventually satisfy the condition of the statement.

□

***Proof of Lemma* 2.4.** Suppose that (X_0', X_1', r') is the configuration right after P_0 plays in the turn. We divide the case into (a) $|X_0'| < |X_0|$ (i.e., P_0 discards some card x), and (b) $|X_0'| = |X_0|$ (i.e., P_0 passes the turn).

We first consider (a). Here, we will show that $\mu(G_1(r')\setminus\min X_1') > \mu(G_0\setminus\min X_0')$ holds. After P_0 discarding x, the turn moves to P_1 and the table card changes from r to x. In this case, P_1 becomes the active player and the configuration graph of P_1 for $(X_0', X_1', r') = (X_0\setminus\{x\}, X_1, x)$ is $G_1(r')\setminus\min X_0' = (X_1, X_0\setminus\{x\}\cup\{x\}\setminus\min(X_0\setminus\{x\})) = (X_1, X_0\setminus\min(X_0\setminus\{x\}))$. If $x \neq \min X_0$, $G_1(r')\setminus\min X_0'$ is exactly same as $G_1\setminus\min X_0 = (X_1, X_0\setminus\min X_0))$; $\mu(G_1(r')\setminus\min X_0') = \mu(G_1\setminus\min X_0)$ holds. If $x = \min X_0$, $G_1(r')\setminus\min X_0' = (X_1, X_0\setminus\min_2 X_0)$, which is the graph obtained by replacing $\min X_2$ of $G_1\setminus\min X_0$ with $\min X_1$; if there is an edge $(x', \min X_2)$ in $G_1\setminus\min X_0$, there is also an edge $(x', \min X_1)$ in $G_1(r')\setminus\min X_0'$. This implies that $\mu(G_1(r')\setminus\min X_0') \geq \mu(G_1\setminus\min X_0)$. Thus in any case, $\mu(G_1(r')\setminus\min X_0') \geq \mu(G_1\setminus\min X_0)$ holds. On the other hand, the configuration graph of non-active player P_0 becomes $G_0\setminus\min X_1' = (X_0', X_1'\setminus\min X_1') = (X_0\setminus\{x\}, X_1\setminus\min X_1)$, which is the graph obtained by deleting x and r with an edge from $G_0(r)\setminus\min X_1 = (X_0, X_1\setminus X_1\cup\{r\})$. Since by deleting a pair of two vertices with an edge from a graph its maximum matching size decreases at least by one, $\mu(G_0(r) \setminus \min X_1) > \mu(G_0 \setminus \min X_1')$ holds. Hence, $\mu(G_1(r') \setminus \min X_0') \geq \mu(G_1 \setminus \min X_0) \geq \mu(G_0(r) \setminus \min X_1) > \mu(G_0 \setminus \min X_1')$ holds for $|X_0'| < |X_0|$.

We next consider case (b). Since P_0 passes the turn, the table becomes empty; 0 is put at the table. Then P_1 becomes the active player and the configuration becomes $(X_0', X_1', r') = (X_0, X_1, 0)$. At this moment, $|X_0'| = |X_0|$ holds, which does not satisfy the required condition of the lemma yet. Thus we show below that P_1 can win or the precondition of lemma holds with fewer cards of P_1 by P_1 appropriately playing, which is reduced to the situation right before (a) and (b) with fewer cards. This implies that case (b) occurs at most $|X_1|$ times, and the situation is eventually reduced to case (a); the lemma holds.

Now the configuration graphs of active P_1 and non-active P_0 are $G_1(r') \setminus \min X_0' = (X_1, X_0 \cup \{0\} \setminus \min X_0)$ and $G_0 \setminus \min X_1' = (X_0, X_1 \setminus \min X_1)$, respectively. We divide the case into (b-1) there is a maximum matching M of $G_1 \setminus \min X_0$ such that it does not touch a vertex x other than $\min X_1$ in X_1, or (b-2) not. In (b-1), let P_1 discard the card x. Then, the configuration becomes $(X_0'', X_1'', r'') = (X_0, X_1 \setminus \{x\}, x)$; the configuration graph of active P_0 changes from $G_0(r) \setminus \min X_1 = (X_0, X_1 \cup \{r\} \setminus \min X_1)$ to $G_0(r'') \setminus \min X_1'' = (X_0, X_1 \setminus \min X_1)$, whose maximum matching size is at most $\mu(G_0(r) \setminus \min X_1)$. On the other hand, the configuration graph of non-active player P_1 changes from $G_1 \setminus \min X_0 = (X_1, X_0 \setminus \min X_0)$ to $G_1 \setminus \min X_0'' = (X_1 \setminus \{x\}, X_0 \setminus \min X_0)$, whose maximum matching size is equal to $\mu(G_1 \setminus \min X_0)$, because the maximum matching M of $G_1 \setminus \min X_0$ is also a maximum matching of $G_1 \setminus \min X_0''$. Thus, $\mu(G_1 \setminus \min X_0'') = \mu(G_1 \setminus \min X_0) \geq \mu(G_0(r) \setminus \min X_1) \geq \mu(G_0(r'') \setminus \min X'')$ holds with $|X_1''| < |X_1|$.

In (b-2), let P_1 discard $\min X_1$, which is always possible by $r' = 0$. If $|X_1| = 1$ (i.e., $\min X_1$ is the last card of P_1), P_1 is the winner. Thus we consider the case $|X_1| \geq 2$. The configuration becomes $(X_0'', X_1'', r'') = (X_0, X_1 \setminus \min X_1, \min X_1)$; the configuration graph of active P_0 changes from $G_0(r) \setminus \min X_1 = (X_0, X_1 \cup \{r\} \setminus \min X_1)$ to $G_0(r'') \setminus \min X_1'' = (X_0, X_1 \setminus \min_2 X_1)$, and the configuration graph of non-active P_1 changes from $G_1 \setminus \min X_0 = (X_1, X_0 \setminus \min X_0)$ to $G_1 \setminus \min X_0'' = (X_1 \setminus \min X_1, X_0 \setminus \min X_0)$. Here note that all the vertices in $X_1 \setminus \min X_1$ are covered by any maximum matching of $G_1 \setminus \min X_0$. We further consider subcases (b-21) $\min X_1$ is also covered by a maximum matching of $G_1 \setminus \min X_0$, (i.e., all the vertices in X_1 is covered by a maximum matching) and (b-22) not. In (b-21), $\mu(G_1 \setminus \min X_0) = |X_1|$ and $\mu(G_1 \setminus \min X_0'') = |X_1| - 1$ hold and $\mu(G_0(r'') \setminus \min X_1'') \leq |X_1 \setminus \min_2 X_1| \leq |X_1| - 1$, which implies $\mu(G_1 \setminus \min X_0'') \geq \mu(G_0(r'') \setminus \min X_1'')$. In (b-22), $\mu(G_1 \setminus \min X_0) = |X_1| - 1$ but $\mu(G_1 \setminus \min X_0'')$ is laso $|X_1| - 1$ because $\min X_1$ is not touched by a maximum matching edge. Also $\mu(G_0(r'') \setminus \min X_1'') \leq |X_1 \setminus \min_2 X_1| \leq |X_1| - 1$, which again implies $\mu(G_1 \setminus \min X_0'') \geq \mu(G_0(r'') \setminus \min X_1'')$. Both the cases satisfy the precondition of the lemma with fewer cards of P_1. This completes the proof. □

Lemma 2.2 implies that the winner decision in TANHINMIN is done by computing μ's, the sizes of maximum matchings. Since the maximum matching size of a graph with n vertices can be computed in polynomial time (e.g., $\mathcal{O}(n^{5/2})$-time [28]), we can determine the winner of an instance of TANHINMIN in polynomial time.

In the rest of this section, we will show that it is easier to compute a maximum matching of a configuration graph due to its transitive property; it does not require a general maximum matching algorithm, and it can be done just by a greedy manner. We show the following lemma.

Lemma 2.5. Suppose that (X_0, X_1, r) are given, where X_0 and X_1 are sorted. Then $\mu(G_0(r) \setminus \min X_1)$ and $\mu(G_1 \setminus \min X_0)$ can be computed in $\mathcal{O}(|X_0| + |X_1|)$ time.

The proof of Lemma 2.5 utilizes the property that an arbitrary bipartite graph $G = (X_0, X_1)$ defined by the strength order has a special edge that can be contained in a maximum matching. We have the following lemma.

Lemma 2.6. If a bipartite graph $G = (X_0, X_1, E(X_0, X_1))$ is connected, there exists a maximum matching that contains edge $(\min X_0, \min X_1)$.

Proof. Since G is connected, $(\min X_0, \min X_1)$ is an edge of G. Suppose that M is a maximum matching that does not contain $(\min X_0, \min X_1)$. If neither $\min X_0$ nor $\min X_1$ is matched in M, it contradicts the maximality of M, so either $\min X_0$ or $\min X_1$ is matched. If $\min X_0$ is matched and $\min X_1$ is not, the matching obtained by replacing matching edge $(\min X_0, x)$ with $(\min X_0, \min X_1)$ in M is also maximum. The case where $\min X_1$ is matched and $\min X_0$ is not can be similarly handled. If both $\min X_0$ and $\min X_1$ are matched but not by $(\min X_0, \min X_1)$, M contains $(\min X_0, x)$ and $(x', \min X_1)$. Since this implies $x' > \min X_0 > x$, G has edge (x', x). Then the matching obtained by replacing $(\min X_0, x)$ and $(x', \min X_1)$ with (x', x) and $(\min X_0, \min X_1)$ has the same cardinality with M, which is also maximum. This complete the proof.

□

By Lemma 2.6, we can easily find an edge contained in a maximum matching by focusing on the connected component. Algorithm 1 implements the idea, and it can efficiently find a maximum matching.

GAME 2 greedy algorithm

1: $G_0 = (X_0, X_1)$ is given, where $X_0 = \{x_1, x_2, \ldots, x_{n_0}\}, x_1 < x_2 < \cdots < x_{n_0}$, and $X_1 = \{y_1, y_2, \ldots, y_{n_1}\}, y_1 < y_2 < \cdots < y_{n_1}$. Set $M := \emptyset, i := 1, j := 1$.

2: If $i > n_0$ or $j > n_1$, then output M and halt. Otherwise, check whether $x_i > y_j$. If yes, then go to Step 3. Otherwise $i := i + 1$ and go to Step 2.

3: Set $M := M \cup \{(x_i, y_j)\}$, and $i := i + 1$ and $j := j + 1$. Go to Step 2.

Proof of Lemma 2.5. We first show the correctness of Algorithm 2. It slides pointers i and j to find the "least" edge to include in M by using Lemma 2.6. Step 1 is the initialization. Step 2 slides pointer i to eliminate isolated vertices in X_0. If x_i has an edge (x_i, y_j), it is included in M in Step 3, and then i and j are updated and go back to Step 2.

The running time of the algorithm is obviously linear, because i and j incrementally increases, and the check of the inequality about x_i and y_j (stored in arrays) can be done in a constant time. $\square$

Lemma 2.5 assumes that X_0 and X_1 are given in a sorted form. We think that it is a natural assumption because TANHINMIN is a card game, and cards in X_0 and X_1 can be sorted in $\mathcal{O}(n)$ time by a non-comparative sorting algorithm; we have Theorem 2.1. In case when $|X_0| + |X_1|$ is much smaller than n, a non-comparative sorting could be slow and a standard comparison-based sort algorithm's $\mathcal{O}((|X_0| + |X_1|)\log(|X_0| + |X_1|))$ would be better.

Chapter 3

TANHINMIN with cut cards

3.1 Introduction

In this chapter, we extend the results in Section 3 to TANHINMIN with extra rule which called “cut rule”. In a popular rule of DAIHINMIN, the rule is also called 8-cut rule, which is named after a special role of 8. In TANHINMIN, the cards with strength are in total preorder, though in TANHINMIN with 8-cut rule they are in a preorder that is almost total except “8-cut card”. In DAIHINMIN with 8-cut rule, once an active player discards “8”, the other players must pass for the “8” even if they have “9” or greater cards than “8”. In the sense, “8” is considered a kind of maximal. The cut rule in this paper is a generalization of the 8-cut rule.

3.2 Definition

To introduce the cut rule, we install cards equipped with the role of “8”. To this end, we introduce “8-cut” cards, or simply cut cards. Let the set of cut cards denote $\{\tilde{1}, \tilde{2}, \ldots, \tilde{n}\}$. In the strength system, $\tilde{i}$ is stronger than $1, 2, \ldots, i-1$, but $\tilde{i}$ and i are not comparable. As same as “8” in DAIHINMIN, no card is stronger than $\tilde{i}$ for any i. For example, if ordinary 5 is at the top of the table, one can discard an ordinary card which is equal to or greater than 6, or a cut card $\tilde{i}$ where i is equal to or greater than 6. On the other hand, if a player discards an arbitrary cut card $\tilde{i}$, the other players must pass the turn, and then the table is cleared.

To model TANHINMIN with cut cards, we introduce additional symbols to the original TANHINMIN. We again use X_0 and X_1 to represent the sets of (ordinary) cards that belong to P_0 and P_1, respectively. Additionally, let Y_0 and Y_1 denote the sets of cut cards that belong to P_0 and P_1, respectively. The configuration of TANHINMIN with cut cards forms (X_0, Y_0, X_1, Y_1, r). Similarly to the basic

TANHINMIN, we construct bipartite graphs $\tilde{G}_0(r) = (X_0 \cup Y_0, X_1 \cup \{r\})$ and $\tilde{G}_1 = (X_1 \cup Y_1, X_0)$.

Like Lemma 2.2, the winner of TANHINMIN with cut cards is determined by the maximum matching sizes of two graphs obtained from $\tilde{G}_0(r)$ and $\tilde{G}_1$, that is, $\mu(\tilde{G}_0(r) \setminus \min X_1)$ and $\mu(\tilde{G}_1 \setminus \min X_0)$, where $\min X_0$ (resp., $\min X_1$) denotes the weakest card of X_0 (resp., X_1).

3.3 Winner decision

We can decide the winning player of 2-player TANHINMIN with cut cards in $\mathcal{O}(n)$ time.

Theorem 3.1. *Given a configuration $(X_0 \cup Y_0, X_1 \cup Y_1, r)$ of 2-player TANHINMIN with cut card with n cards, we can decide the winning player in $\mathcal{O}(n)$ time.*

We can prove Theorem 3.1 by proving several lemmas as Theorem 2.1. Lemma 2.2 corresponds to Lemma 3.4. To handle exceptional cases, we use Lemma 3.3. Here, we just concentrate on explaining major differences to avoid repetitions.

Similarly to the ordinary TANHINMIN, the values of $\mu(\tilde{G}_0(r) \setminus \min X_1)$ and $\mu(\tilde{G}_1 \setminus \min X_0)$ represent the essential information on whether or not to have a winning strategy, but it is not all. There is a situation where we cannot decide only by $\mu(\tilde{G}_0(r) \setminus \min X_1)$ and $\mu(\tilde{G}_1 \setminus \min X_0)$, which occurs when a cut card in hand is not strong enough. We roughly name a "*surplus*" state such as the state that a player has cut cards that can be discarded only on empty table or the weakest card that the opponent player has.

For example, suppose that P_0 has an ordinary card 2 and a cut card $\tilde{1}$, P_1 has only ordinary cards of 1 and 3 (no cut card), and a card on the table is 1 (Fig. 3.1 (a)). In this case, P_0 has only two options: to select pass or to discard 2. If P_0 discards 2 and the turn moves P_1, P_1 can discard 3 (Fig. 2 (c)). In the other case, P_0 selects pass and the turn moves P_1. Then P_1 can also discard 3 (Fig. 2 (d)). Both cases P_1 can discard 3 and the turn moves to P_0 (Fig. 2 (e)). He/she has only one option: to select pass and the turn moves to P_0. In P_1's turn, P_1 can discard 1 and P_1 wins (Fig. 2 (f)). In this example, $\mu(\tilde{G}_0(r) \setminus \min X_1) = 1$ and $\mu(\tilde{G}_1 \setminus \min X_0) = 0$ hold. If we apply the decision rule like Lemma 2.2, P_0 seems to have a winning strategy, but it is not correct, as seen above. To handle such an exceptional situation, we introduce the name *surplus*, and we say that P_0 is in the *surplus state*. In fact, P_0 can discard $\tilde{1}$ once the turn comes in empty table. We define such a status as follows:

Definition 3.2. Suppose P_0 and P_1 are active and non-active, respectively. We say that active P_0 is in a *surplus status* if all of the following conditions are satisfied:

(a) initial setting

(b) P_0 discards 2.

(c) P_1 discards 3. (after P_0 discards 2.)

(d) P_1 discards 3. (after P_0 selects pass.)

(k) The table is cleared.

(l) P_1 wins the game.

Figure 3.1: A play example of 2-player TANHINMIN with cut card

$Y_0 \neq \emptyset$, $\mu(\tilde{G}_1) = |X_0|$, $r \notin N(Y_0)$ and $|N(Y_0)| \leq 1$. Similarly, we say that non-active P_1 is in a *surplus status* if all of the following conditions are satisfied: $Y_1 \neq \emptyset$, $\mu(\tilde{G}_0) = |X_1 \cup \{r\}|$, $\max(X_0 \cup Y_0) \notin N(Y_1)$ and $|N(Y_1)| \leq 1$.

Although we omit the detailed proof, only one player can be in a surplus status simultaneously. If the opponent player is in a surplus status, the player is the winner.

Lemma 3.3. Given a configuration of 2-player TANHINMIN with cut cards, if non-active P_1 (resp., active P_0) is in the surplus status, active P_0 (resp., P_1) has a winning strategy.

We can prove this lemma by a mathematical induction with respect to $|X_0|$ and $|X_1|$, which are similar to the one of Lemma 2.2. We omit the detail to avoid tedious repetitions.

In the other cases, we can also determine which player has a winning strategy using maximum matching size as explained in the following lemma. By applying this and the arguments like Lemma 2.5, we obtain Theorem 3.1.

Lemma 3.4. Given a configuration $(X_0 \cup Y_0, X_1 \cup Y_1, r)$ of 2-player TANHINMIN with cut cards, where P_0 and P_1 respectively are active and non-active, and neither of them is in a surplus status, P_0 has a winning strategy when $\mu(\tilde{G}_0(r) \setminus \min X_1) > \mu(\tilde{G}_1 \setminus \min X_0)$ holds, and P_1 has a winning strategy otherwise.

Proof of lemma 3.4. We prove these lemmas by a mathematical induction with respect to $|X_0|$ and $|X_1|$.
(Base step)

We consider the case where $|X_0| = 0$ holds: if $|Y_1| > 0$ and $\mu(\tilde{G}_0(r)) = \mu(\tilde{G}_0(r) \setminus \{x_1(i)|0 \leq i \leq |X_1| - \mu(\tilde{G}_0(r))\}$ and $\min(X_1 \setminus \min X_1) > x_0(|X| - \mu(\tilde{G}_1)$ and $\max(X_0 \cup Y_0) \notin N(Y_1)$ and $|N(Y_1)| \leq 1$ hold, P_0 wins. Note that $|X_0 \cup Y_0| = |Y_0|$ holds because$|X_0| = 0$ holds and $\{r\} \in N(Y_0)$ holds because $\mu(\tilde{G}_0(r)) = \mu(\tilde{G}_0(r) \setminus \{x_1(i)|0 \leq i \leq |X_1| - \mu(\tilde{G}_0(r))\}$ holds. Since $\{r\} \in N(Y_0)$ holds, P_0 can discard a card $y \in Y_0$. If $|Y_0| = 1$ holds, P_0 just discards the unique card and wins. When $|Y_0| > 1$, P_0 discards a card $y \in Y_0$, then y becomes the table card instead of r, and $G_1(r)$ changes to $G_1(r) \setminus r = G_1$; P_1 cannot discard any card because a card on the table is cut-cards. Therefore P_1 must select "pass". Then the turn moves and P_0 becomes the active player with an empty table, which means r is replaced by 0; That is, $|X_0 \cup Y_0|$ decreases by 1. Thus the situation is recursively reduced to $|X_0 \cup Y_0| = 1$ and P_0 eventually wins.

Next, we consider the case where $X_1 = 0$ holds: if when $|Y_0| > 0$ and $min(X_1 \setminus minX_1) > x_0(|X| - \mu(\tilde{G}_1)$ and $min(X_1 \setminus minX_1) > x_0(|X| - \mu(\tilde{G}_1)$ and $r \notin N(Y_0)$ and $|N(Y_0)| \leq 1$ hold hold, P_1 wins. Since $r \notin N(Y_0)$ holds, P_0 discard ordinary card $x \in X_0$ or pass. If P_0 discard ordinary card $x \in X_0$, then the turn moves and P_1 becomes the active player with an empty table, which means r is replaced by x; In the other case: when P_0 selects pass, then the turn moves and P_1 becomes the active player with an empty table, which means r is replaced by 0

In both of these cases, $|X_1| = 0$ and $|Y_0| > 0$ and $\mu(\tilde{G}_1) = \mu(\tilde{G}_1 \setminus \{x_0(i)|0 \leq i \leq |X_1| - \mu(\tilde{G}_1))\}$ and $\max\{X_1 \cup Y_1\} \notin N(Y_1)$ and $|N(Y_0)| \leq 1$ hold, so P_0 has a winning strategy.

(Induction step) Assume that for $|X_0| \leq k$, P_0 has a winning strategy if $|Y_1| > 0$ and $\mu(\tilde{G}_0(r)) = \mu(\tilde{G}_0(r) \setminus \{x_1(i)|0 \leq i \leq |X_1| - \mu(\tilde{G}_0(r))\}$ and $min(X_0 \setminus minX_0) > x_1(|X| - \mu(\tilde{G}_0(r))$ and $|N(Y_1)| \leq 1$ hold. And assume that for $|X_1 \cup Y_1| \leq k$, P_1 has a winning strategy $|Y_0| > 0$ and $\mu(\tilde{G}_1) = \mu(\tilde{G}_1 \setminus \{x_0(i)|0 \leq i \leq |X_1| - \mu(\tilde{G}_1))\}|$ and $\min(X_1 \setminus minX_1) > x_0(|X| - \mu(\tilde{G}_1)$ and $r \notin N(Y_0)$ and $|N(Y_0)| \leq 1$ hold. We will show that for $X_0 = k + 1$, P_0 has a winning strategy if $|Y_1| > 0$ and $\mu(\tilde{G}_0(r)) = \mu(\tilde{G}_0(r) \setminus \{x_1(i)|0 \leq i \leq |X_1| - \mu(\tilde{G}_0(r))\}$

and $min(X_1 \setminus \min X_1) > x_0(|X| - \mu(\tilde{G_1})$ and $|N(Y_1)| \leq 0$ hold, P_0 has a winning strategy(i); And show that for P_1 has a winning strategy if $|Y_0| > 0$ and $\mu(\tilde{G_1}) = |X_0 \setminus \{x_0(i)|0 \leq i \leq |X_0| - \mu(\tilde{G_1})\}|$ and $r \notin N(Y_0)$ and $|N(Y_0)| \leq 1$ hold(ii).

We first show (i) Since $\mu(\tilde{G_0}(r)) = \mu(\tilde{G_0}(r) \setminus \{x_1(i)|0 \leq i \leq |X_1| - \mu(\tilde{G_0}(r))\}$ holds, P_0 has a card to discard on the table card. Therefore P_0 can discard a card $x \in X_0$ or card $y \in Y_0$. Consider two cases: where $\min(x, y) > r|x \in X_0 \setminus \min X_0, y \in Y_0$ is certain $x \in x_0$(case α) or not(case β).

First we consider case α. In this case P_0 discard min $x > r|x \in X_0 \setminus \min X_0$. After P_0 discard a card $x \in X_0$, P_1 becomes active player and r is replaced by x. Now, $|X_0| \leq k$ and $|Y_1| > 0$ and $\mu(\tilde{G_0}) = \mu(\tilde{G_0} \setminus \{x_1(i)|0 \leq i \leq |X_0| - \mu(\tilde{G_0}))\}|$ and $\min(X_0 \setminus \min X_0) > x_1(|X| - \mu(\tilde{G_0})$ and $r \notin N(Y_1)$ and $|N(Y_1)| \leq 1$ hold. Consequently, case α reduced.

Next, we consider case β. In this case, P_0 discard min $y > r|y \in X_0 \setminus \min X_0$ and P_1 becomes the active player with table card y. P_1 cannot discard any card, so P_0 becomes the active player with empty table. In this case, P_0 discard min $x > 0|x \in X_0 \setminus \min X_0$ and P_1 becomes active player with table card x. Now, $|X_0| \leq k$ and $|Y_1| > 0$ and $\mu(\tilde{G_0}(r)) = \mu(\tilde{G_0}(r) \setminus \{x_1(i)|0 \leq i \leq |X_1| - \mu(\tilde{G_0}(r))\}$ and $\min(X_0 \setminus \min X_0) > x_1(|X| - \mu(\tilde{G_0}(r))$ and $|N(Y_1)| \leq 1$ hold. Consequently, case β also reduced.

Next, we show (ii). Since $r \notin N(Y_0)$ holds, P_0 has no cut card to play on the table. Therefore P_0 has only two option: discarding any card $x \in X_0$(ii) or selecting "pass"(ii). Either P_0 play a card on the table or selecting "pass", after that, $\mu(\tilde{G_1}(r)) = \mu(\tilde{G_1}(r) \setminus \{x_0(i)|0 \leq i \leq |X_0| - \mu(\tilde{G_1}(r))\}$ holds since $min(X_1 \setminus minX_1) > x_0(|X| - \mu(\tilde{G_1})$ holded previous P_0's turn. Now, $|Y_1| > 0$ and $\mu(\tilde{G_0}(r)) = \mu(\tilde{G_0}(r) \setminus \{x_1(i)|0 \leq i \leq |X_1| - \mu(\tilde{G_0}(r))\}$ and $min(X_1 \setminus minX_1) > x_0(|X| - \mu(\tilde{G_1})$ and $|N(Y_1)| \leq 0$ hold. This is case (i). Consequently (ii) reduced (i). $\square$

Now that we prepare to prove lemma 3.3, we show the proof.

Proof. We prove these lemmas by a mathematical induction with respect to $|X_0|$ and $|X_1|$.

(Base step)

We consider the case where $|X_0| = 0$ holds: if $|Y_1| > 0$ and $\mu(\tilde{G_0}(r)) = |X_1 \cup \{r\}|$ and $\max(X_0 \cup Y_0) \notin N(Y_1)$ and $|N(Y_1)| \leq 1$ hold, P_0 wins. Note that $|X_0 \cup Y_0| = |Y_0|$ holds because$|X_0| = 0$ holds and $\{r\} \in N(Y_0)$ holds because $\mu(\tilde{G_0}(r)) = |X_1 \cup \{r\}|$ holds.

Since $\{r\} \in N(Y_0)$ holds, P_0 can discard a card $y \in Y_0$. If $|Y_0| = 1$ holds, P_0 just discards the unique card and wins. When $|Y_0| > 1$, P_0 discards a card $y \in Y_0$, then y becomes the table card instead of r, and $G_1(r)$ changes to $G_1(r) \setminus r = G_1$;

P_1 cannot discard any card because a card on the table is cut-cards. Therefore P_1 must select "pass". Then the turn moves and P_0 becomes the active player with an empty table, which means r is replaced by 0; That is, $|Y_0|$ decreases by 1. Thus the situation is recursively reduced to $|Y_0| = 1$ and P_0 eventually wins.

Next, we consider the case where $X_1 = 0$ holds: if when $|Y_0| > 0$ and $\mu(\tilde{G_1}) = |X_0|$ and $r \notin N(Y_0)$ and $|N(Y_0)| \leq 1$ hold hold, P_1 wins. Since $r \notin N(Y_0)$ holds, P_0 discard ordinary card $x \in X_0$ or pass. If P_0 discard ordinary card $x \in X_0$, then the turn moves and P_1 becomes the active player with an empty table, which means r is replaced by x; In the other case: when P_0 selects pass, then the turn moves and P_1 becomes the active player with an empty table, which means r is replaced by 0

In both of these cases, $|X_1| = 0$ and $|Y_0| > 0$ and $\mu(\tilde{G_1}) = |X_1|$ and $\max\{X_1 \cup Y_1\} \notin N(Y_1)$ and $|N(Y_0)| \leq 1$ hold, so P_0 has a winning strategy.

(Induction step)

Assume that for $|X_0| \leq k$, P_0 has a winning strategy if $|Y_1| > 0$ and $\mu(\tilde{G_0}(r)) = |X_1 \cup \{r\}|$ and $|N(Y_1)| \leq 1$ hold. And assume that for $|X_1 \cup Y_1| \leq k$, P_1 has a winning strategy $|Y_0| > 0$ and $\mu(\tilde{G_1}) = |X_0|$ and $r \notin N(Y_0)$ and $|N(Y_0)| \leq 1$ hold.

We will show that for $X_0 = k + 1$, P_0 has a winning strategy if $|Y_1| > 0$ and $\mu(\tilde{G_0}(r)) = |X_1 \cup \{r\}|$ and $|N(Y_1)| \leq 0$ hold, P_0 has a winning strategy(i); And show that for P_1 has a winning strategy if $|Y_0| > 0$ and $\mu(\tilde{G_1}) = |X_0|$ and $r \notin N(Y_0)$ and $|N(Y_0)| \leq 1$ hold(ii).

We first show (i) Since $\mu(\tilde{G_0}(r)) = |X_1 \cup \{r\}|$ holds, P_0 has a card to discard on the table card. Therefore P_0 can discard a card $x \in X_0$ or card $y \in Y_0$. Consider two cases: where $\min(x, y) > r | x \in X_0 \setminus \min X_0, y \in Y_0$ is certain $x \in x_0$(case α) or not(case β).

First we consider case α. In this case P_0 discard min $x > r | x \in X_0 \setminus \min X_0$ After P_0 discard a card $x \in X_0$, P_1 becomes active player and r is replaced by x. Now, $|X_0| \leq k$ and $|Y_1| > 0$ and $\mu(\tilde{G_1}) = |X_0|$ and $r \notin N(Y_1)$ and $|N(Y_1)| \leq 1$ hold. Consequently, case α reduced.

Next, we consider case β. In this case, P_0 discard min $y > r | y \in X_0 \setminus \min X_0$ and P_1 becomes the active player with table card y. P_1 cannot discard any card, so P_0 becomes the active player with empty table. In this case, P_0 discard min $x > 0 | x \in X_0 \setminus \min X_0$ and P_1 becomes active player with table card x. Now, $|X_0| \leq k$ and $|Y_1| > 0$ and $\mu(\tilde{G_1}) = |X_0|$ and $|N(Y_1)| \leq 1$ hold. Consequently, case β also reduced.

Next, we show (ii). Since $r \notin N(Y_0)$ holds, P_0 has no cut card to play on the table. Therefore P_0 has only two option: discarding any card $x \in X_0$(ii) or selecting "pass"(ii). If P_0 play a card on the table, after that $|X_1| = k + 1$ and $|Y_0| > 0$ and $\mu(\tilde{G_1}) = |X_0|$ and $r \notin N(Y_0)$ and $|N(Y_0)| \leq 1$ hold. If P_0

selecting “pass”, after that, $|Y_1| > 0$ and $\mu(\tilde{G}_0(r)) = \mu(\tilde{G}_0(r) \setminus \{x_1(i)|0 \leq i \leq |X_1| - \mu(\tilde{G}_0(r))\}$ and $min(X_1 \setminus minX_1) > x_0(|X| - \mu(\tilde{G}_1)$ and $|N(Y_1)| \leq 0$ hold. This is case (i). Since lemma 3.4, P_1 has a winning strategy. Consequently (ii) reduced (i).

$\square$

Chapter 4

TANHINMIN with imperfect information

4.1 Introduction

TANHINMIN is a simplified and perfect information variant of DAIHINMIN game, which is major playing card game in Japan. It is proved that it can be decided in linear time which player has a winning strategy in 2-player TANHINMIN game in Chapter 2. In this chapter, we concern with how we obtain a winning strategy for the imperfect information variant of TANHINMIN game. If any information about the opponent player's hand is not given at all, it is obviously difficult to find a winning strategy, though such a hard situation does not likely happen in real game plays; players usually receive some little information about the opponent player's hand through a game, e.g., the number of cards. To handle the situation that a player can receive some information about the opponent player's hand, we introduce an oracle model in which the oracle provides partial information about the opponent's hand. Interestingly, when players can get partial information of the opponents' hands via oracle, the winning player can find a winning strategy as if it is the (perfect information) TANHINMIN. Furthermore, we show various results about other relationships between the power of oracles and the existence of a computable winning strategy.

4.2 Graph Model of TANHINMIN

We assume basic knowledge of graph theory. Let $G = (V, E)$ be a graph, where V is the set of vertices and E is the set of edges.

All the graphs that we consider in this paper are *bipartite*, that is, there is a bipartition (V_0, V_1) of V such that $E \subseteq \{(p, q) \mid p \in V_0, q \in V_1\}$. To specify the bipartition, we denote $G = (V_0, V_1, E)$ instead of $G = (V, E)$. For graph G and a vertex v of G, $N_G(v)$ denotes the set of neighboring vertices to v in G, that is, $N_G(v) = \{u \in V \mid \{u, v\} \in E\}$. We sometimes use notation $N(v)$ instead of $N_G(v)$ if the graph that we consider is clear. For $S \subseteq V$, $N(S)$ similarly denotes the set of vertices neighboring to any vertex in S, that is, $N(S) = \bigcup_{v \in S} N(v)$. For graph $G = (V, E)$ and $v \in V$, let $G \setminus v$ denote a graph obtained by deleting v and its incident edges. For a graph $G = (V, E)$, a subset M of E is called *matching* if no two edges in M share an end. For a graph G, we denote the size of a maximum matching by $\mu(G)$.

Suppose that the two players of our TANHINMIN are P_0 and P_1, where P_0 is the active player and P_1 is the non-active player. We first fix a turn to consider. At the turn, we respectively denote by X_0 and X_1 the cards belonging to P_0 and P_1, and by $\{r\}$ the top card on the table. These provide sufficient information to describe the situation of the turn; triplet (X_0, X_1, r) define the configuration of the turn.

Note that in a play of TANHINMIN cards on table are sometimes cleared, and then $\{r\}$ is empty. In such a case, we virtually consider that 0 is at the top of the cards on table. For example, in Figure 1, $X_0 = \{1, 3, 4, 5, 8\}$, $X_1 = \{2, 6, 7\}$ and $r = 0$ at (a), and $X_0 = \{2, 6, 7\}$, $X_1 = \{3, 4, 5, 8\}$ and $r = 1$ right after (b).

We then give a graph model of TANHINMIN; for a configuration, we construct several graphs.

The vertices correspond to cards in $X_0 \cup X_1 \cup \{r\}$, and use the same symbols to represent them. For configuration (X_0, X_1, r), we then construct graphs G_0 and $G_0(r)$ as follows:

$$\begin{aligned} G_0 &= (X_0, X_1, E_0), \\ \text{where } E_0 &= \{(i, j) \mid i \in X_0, j \in X_1, i > j\}, \\ G_0(r) &= (X_0, X_1 \cup \{r\}, E_0), \\ \text{where } E_0 &= \{(i, j) \mid i \in X_0, j \in X_1 \cup \{r\}, i > j\}. \end{aligned}$$

Similarly, we define

$$\begin{aligned} G_1 &= (X_1, X_0, E_1), \\ \text{where } E_1 &= \{(i, j) \mid i \in X_0, j \in X_1, j > i\}. \end{aligned}$$

Here, graph $G_0(r)$ represents which cards P_0 can discard for cards in $X_1 \cup \{r\}$. Graph G_1 represents which cards P_1 can discard for cards in X_0. If $X_0 = \emptyset$ or $X_1 = \emptyset$, P_0 or P_1 is obviously the winning player, respectively. Thus we assume that both X_0 and X_1 are nonempty in the following.

As we see below in Proposition 4.1, the winner of TANHINMIN is determined by the maximum matching sizes of two graphs obtained from $G_0(r)$ and G_1, that is, $\mu_0 \stackrel{\text{def}}{=} \mu(G_0(r) \setminus \min X_1)$ and $\mu_1 \stackrel{\text{def}}{=} \mu(G_1 \setminus \min X_0)$, where $\min X_0$ (resp., $\min X_1$) denotes the weakest card of X_0 (resp., X_1). Since these graphs play important roles in the winner decision, we name $G_0(r) \setminus \min X_1$ and $G_1 \setminus \min X_0$ the configuration graph of active player P_0 and the configuration graph of active player of non-active player P_1, respectively.

Proposition 4.1. (Lemma 2.2) Given a configuration (X_0, X_1, r) of 2-player TANHINMIN with n cards, P_0 has a winning strategy when $\mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0)$, and P_1 has a winning strategy otherwise.

Based on this proposition, the winner of a given perfect information TANHINMIN can be computed in linear time, and it also gives an insight that the winning strategy is strongly related to the maximum matching structures of the configuration graphs. In the following, we call P_0 (resp., P_1) a player satisfying the winning inequality if $\mu_0 > \mu_1$ (resp., $\mu_0 \leq \mu_1$). By these, our oracle-based analyses of imperfect information variants of TANHINMIN also utilize the configuration graphs and their maximum matching.

4.3 Imperfect Information TANHINMIN with structural oracles

As we see in the previous section, the winner of 2-player perfect information variant can be computed efficiently, but of course, the perfect information setting is not always realistic, as DAIHINMIN is an imperfect information game in fact. Thus we consider to extend the analyses for the perfect variant to imperfect variants. If "imperfect" means no information, what we can do seems to be nothing, On the other hand, the setting "no information" is quite rare in real game playing situations; players can get some information of their opponents' hands, e.g., how many cards she has, whether she has a specific card, and so on. For example, suppose that we use standard playing cards for DAIHINMIN, which is played in the hidden manner. In spite that it is played in the hidden manner, if a player has four Q cards in the hand, she knows that the other players have no Q. Alternatively, if a player has three Q cards, she knows that there is a player having one Q card. In other words, the DAIHINMIN is rather a partial information game than a game with no information.

Here, it is important to precisely model or control the partial information that the players can receive. In this paper, we introduce a *structural oracle* (or simply call *oracle*) that gives such information.

Here, we formally define a structural oracle. Player 0 (resp., 1) knows her own hand X_0 (resp., 1) and can access an oracle f, which is a function from (X_0, X_1, r) to a certain range. In this paper, we consider two types of oracles. One is called a cardinality oracle, which returns the size of $|X_i|$ $(i = 0, 1)$, the other is a matching size oracle, which returns μ_0 and/or μ_1. Since these values are regarded as functions, they also refer to oracles. For example, $|X_0|$ refers to the oracle that returns $|X_0|$. Note that the number of cards which the opponent player has is a typical information that can be easily obtained during a play of DAIHINMIN, and the cardinality oracles model this. Remind that μ_0 is $\mu(G_0(r) \setminus \min X_1)$ and μ_1 is $\mu(G_1 \setminus \min X_0)$. Also recall that in the 2-players perfect information TANHINMIN, the winner can be determined by computing μ_0 and μ_1 (in Chapter 2, we showed).

We can show the following theorems. All the theorems are about the 2-players TANHINMIN played in the hidden manner, but each player can access some oracles.

(Induction step)

Theorem 4.2. Assume that P_0 and P_1 can access $|X_1|$ and $|X_0|$ oracles. When $|X_1| \leq 2$ and $\mu_0 > \mu_1$ (resp., $|X_0| \leq 2$ and $\mu_0 \leq \mu_1$), P_0 (resp., P_1) has the winning strategy.

Theorem 4.2 implies that there are situations that only the cardinality oracle is strong enough to get information for the winning player. At the same time, the power of the cardinality oracle is very limited to the situation; the size itself is crucial as seen in the next theorem.

Theorem 4.3. Assume that P_0 and P_1 can access $|X_1|$ and $|X_0|$ oracles. Even when $\mu_0 > \mu_1$ (resp., $\mu_0 \leq \mu_1$), there is a game with $|X_1| \geq 3$ (resp., $|X_0| \geq 3$) where P_0 (resp., P_1) cannot take her winning strategy.

It is interesting that Theorems 4.2 and 4.3 may give a guideline to play in real game playing situations, that is, each player can know the number of cards of the other, if the number of cards of the opponent player is 1 or 2, then the player can play as if she knows the opponent's hand; otherwise, some uncertainty remains.

Theorem 4.4. Assume that P_0 and P_1 can access either μ_0 or μ_1 oracles at a certain timing. Player P_0 (resp., P_1) has the winning strategy when $\mu_0 > \mu_1$ ($\mu_0 \leq \mu_1$) at the timing.

Theorems 4.3 and 4.4 contrast well. Theorem 4.5 implies that once a player satisfying the winning inequality can access either μ_0 or μ_1 at some moment, she can perform the best move as if she plays perfect information game. Note that in the setting, a player satisfying the winning inequality cannot identify that she herself is a player satisfying the winning inequality. Thus under the matching oracle, what each player can do is to play as she is a player satisfying the winning inequality. The following theorem shows that the winning scenario is also essential.

Theorem 4.5. Assume that P_0 and P_1 can access all the oracles of $\mu_0, \mu_1, |X_0|$ and $|X_1|$ at some timing t. Even when $\mu_0 > \mu_1$ (resp., $\mu_0 \leq \mu_1$) of timing t changes $\mu_0 \leq \mu_1$ (resp., $\mu_0 > \mu_1$) at some later timing of the game, there is a case with where P_1 (resp., P_0) cannot take her winning strategy.

Corollary 4.6. Assume that player P_0 (resp., P_1) can access oracle either μ_0 or μ_1 every turn. If there is a timing that P_0 (resp., P_1) becomes a player satisfying the winning inequality, P_0 (resp., P_1) wins.

Theorem 4.5 and Corollary 4.6 may contrast. Theorem 4.5 implies that even if a player becomes a player satisfying the winning inequality, the player may not be able to win if the timing is later than the oracle access. Corollary 4.6 implies that if a player can access the matching size oracle every time, she can adjust her strategy to the winning one.

Figure 4.1 summarizes the results.

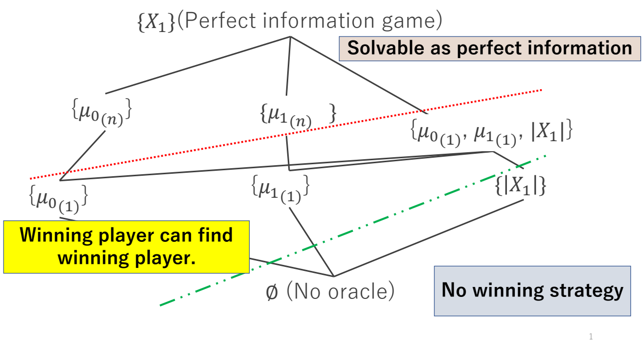


Figure 4.1: Relationship between accessible oracles and solvability of the winner decision

In Figure 4.1, braces show the set of oracles that P_0 can access. For example, $\{\mu_{0(1)}, \mu_{1(1)}, |X_1|\}$ represents that P_0 can access μ_0, μ_1 and $|X_1|$ at the beginning of the game, where $\mu_{0(1)}$'s (1) represents once, as explained below. The top one $\{X_1\}$ represents that the case when P_0 can access X_1 itself, which is equivalent to the perfect information variant. For matching size oracles, how often P_0 can access μ_0 or μ_1 is important. Here, $\mu_{*(1)}$ represents the case in which P_0 can access μ_* once, and $\mu_{*(n)}$ represents the game in which P_0 can access μ_* anytime.

4.4 Proofs

In this section, we give proofs of the theorems shown in Section 4.3.

Proof of Theorem 4.2. We show P_0 can win to play Algorithm 3 when $|X_1| \leq 2$ and $\mu_0 > \mu_1$ holds.by considering the case $|X_1| = 1$ (i) and $|X_1| = 2$(ii).

Algorithm 3 Winning Strategy

1: When P_0 can discard weakest card during P_0's turn (empty table, etc.), discard the second weakest card into play, otherwise the weakest card that can be discarded.

(i)$|X_1| = 1$

We see the case $|X_1| = 1$ and let x_0 be a card (vertex) in X_0. We consider the case where $\mu(G_0(r) \setminus \min X_1) > \mu(G_1 \setminus \min X_0)$ holds and P_0 is the active player. We will show that P_0 wins by using Algorithm 3 in this case.

When $|X_1| = 1$ holds, $\mu(G_0(r) \setminus \min X_1) = 1$ and $\mu(G_1 \setminus \min X_0) = 0$ holds because the value of $\mu(G_0(r) \setminus \min X_1)$ is 1 at most and the value of $\mu(G_1 \setminus \min X_0)$ is 0 at least. If there is no card $\{x_0 | x_0 > r\}$, $\mu(G_1 \setminus \min X_0) = 0$ does not hold because $\mu(G_1 \setminus \min X_0) = \mu(G_1 \setminus \min X_0) = \mu(\emptyset) = 0$. Therefore, there is a card $\{x_0 | x_0 > r\}$ in P_0's hand. P_0 discard a card except for a card $\min X_0$; P_1 must pass since there is no card in P_1's hand and P_0 become the active player with empty table. After that P_0 can repeat discarding non-weakest card and P_1 must pass. Finally P_0 discard the last card in her hand and P_0 can win.

(ii)$|X_1| = 2$

Next, we show P_0 can win to play Algorithm 3 when $|X_1| = 2$ and $\mu_0 > \mu_1$ holds. Since the value of μ_0 is 2 at most when $|X_1| \leq 2$, $\mu_0 = 1$ and $\mu_1 = 0$ or $\mu_0 = 2$ $\mu_1 = 0$ or 1 holds.

At first, we consider the case $\mu_0 = 1$ and $\mu_1 = 0$ holds. Since $\mu_1 = 0$ holds, there is no card in P_1's hand can discard onto table card which is discarded by P_0 unless $\min X_0$.

If there is a card $x_0 > r$ in P_1's hand then discard a card except for a card $\min X_0$. P_1 must pass since there is no card in P_1's hand and P_0 become the active player with empty table. After that P_0 can repeat discarding non weakest card and P_1 must pass. Finally P_0 discard the last card in her hand and P_0 can win.

If there is no card $x_0 > r$ in P_1's hand then P_0 must select pass and P_1 become the active player. P_1 discard any card, P_0 can discard a non-weakest card. P_1 must pass since there is no card in P_1's hand and P_0 become the active player with empty table. After that P_0 can repeat discarding non weakest card and P_1 must pass. Finally P_0 discard the last card in her hand and P_0 can win.

Next, we consider the case $\mu_0 = 2$ and $\mu_1 \leq 1$ holds. Since $\mu_0 = 2$ holds, there is a card in P_0's hand can discard onto table card r. P_0 can discard a non-weakest card as weak as she can play and P_1 become the active player. P_1 may discard a card onto table card or select pass.

If P_1 select pass, P_0 become the active player with empty table after that P_0 can discard a non-weakest card as weak as she can play. This repeats until P_1 discard a card or P_0 win.

If P_1 select pass, P_0 can discard onto a table card which P_1 discards because P_0 can save stronger card than P_1 discards one. After that P_0 can repeat discarding non weakest card and P_1 must pass. Finally P_0 discard the last card in her hand and P_0 can win. □

Next, we give a proof of Theorem 4.3.

Proof of Theorem 4.3. We prove by describing the concrete game which player cannot obtain winning strategy. Let the table be empty and P_0 be turn player. Let the initial hand-set of P_0 be $\{2, 3\}$. In this case, we can mentioned $\{1, 2, 5\}$ or $\{2, 3, 3\}$ as the one of losing hand of P_1 if this game is perfect information; the following shows that even if the hand of P_1 is limited to either of these two, the winning strategy of P_0 differs depending on which one is.

At first, we consider the case where $X_1 = \{1, 2, 5\}$ holds. In this case if P_0 discard a card 2, P_0 can win regardless P_1's after playing: discard a card 5 or pass.

On the other case: if P_1 discard a card 3, then 3 becomes the table card instead of 0, and P_1 becomes active player. P_1 can discard a card 5; In this case P_1 discard 5. Then 5 becomes the table card instead of 3, and P_0 becomes active player. P_1 cannot discard any card because there is no stronger card in her hand; In this case P_0 must be pass. After that, P_1 becomes the active player with an empty table, which means 5 is replaced by 0; P_1 can discard 2. then a card 2 becomes the table card instead of a card 2, and P_0 becomes active player. P_1 cannot discard any card because there is no stronger card in her hand; In this case P_0 must be pass. After that, P_1 becomes the active player with an empty table, which means 2 is replaced by 0; P_1 can discard 0. Finally, P_1 can discard a card 1 and P_1 win. Therefore, P_0 must have winning strategy to discard a card 2 at first, and may lose to discard a card 3 at first.

Next, we consider the case where $X_1 = \{2, 3, 3\}$ holds. In this case if P_0 discard a card 3, P_0 can win regardless P_1's after playing: discard a card 5 or pass.

On the other case: if P_1 discard a card 2, then 2 becomes the table card instead of 0, and P_1 becomes active player. P_1 can discard a card 3; In this case P_1 discard 3. Then 3 becomes the table card instead of 2, and P_0 becomes active player. P_1

cannot discard any card because there is no stronger card in her hand; In this case P_0 must be pass. After that, P_1 becomes the active player with an empty table, which means 3 is replaced by 0; P_1 can discard 3. then a card 3 becomes the table card instead of a c, an empty table P_0 becomes active player. P_1 cannot discard any card because there is no stronger card in her hand; In this case P_0 must be pass. After that, P_1 becomes the active player with an empty table, which means 3 is replaced by 0; P_1 can discard 2. Finally, P_1 can discard a card 1 and P_1 win. Therefore, P_0 must have winning strategy to discard a card 3 at first, and may lose to discard a card 2 at first in the case where $X_1 = \{2, 3, 3\}$ holds.

If each player does not receive any information about the opponent player, P_0 does not have a winning strategy in this game because the above two games are indistinguishable. On the other hand, P_1 also does not have a winning strategy because it may loses depending on the choice of P_0. From the above, this theorem is shown. □

Finally, we prove Theorem 4.5.

Proof of Theorem 4.5. We prove by describing the concrete game in which player cannot obtain winning strategy. Let the initial handset of P_0 be $\{2, 3\}$, initial table card is empty and P_1 is active player. initial player given a oracle that $\mu_0 = 2$, $\mu_1 = 1$, and $|X| = 5$. In this case, we can mentioned $\{1, 1, 2, 4, 5\}$ or $\{1, 2, 3, 3, 4\}$ as the one of losing hand of P_1 if this game is perfect information; The following shows that even if the hand of P_1 is limited to either of these two, the winning strategy of P_0 differs depending on which one is.

Given the following progress in this game:

1. P_1 discards a card 4.
2. Since there is no card to discard for P_0, P_0 selects the pass.
3. P_1 discards a card 1.

Then the card of 1 becomes the table card instead of 0, and P_0 becomes active player. In this case, P_0's hand-set is $\{2, 3\}$, and P_1's hand-set is $\{1, 2, 5\}$ or $\{2, 2, 3\}$ and table card is 1. This situation is equal to the theorem 4.3.

From the above, this theorem is shown. □

Chapter 5

Open-Hand BABANUKI

5.1 Introduction

BABANUKI is a popular game with playing cards in Japan. The basic rule of BABANUKI is quite simple, and many similar games are played all over the world. For example, it is similar to “Old maid" game.

In this chapter, we analyze BABANUKI game as a multi-player discarding game with perfect information by non preference analysis. We newly introduce open-hand BABANUKI based on the ordinary BABANUKI. It is played in the manner that players play BABANUKI with cards faced up. This makes the game a perfect information game, and it becomes worth considering optimal strategies; we consider the winning strategy of open-hand variant BABANUKI. Although the 2-players case is almost obvious, the 3-players case is not, and we give a necessary and sufficient condition of the existence of the winning strategy. Furthermore, for 4-players case, there is a configuration where an endless-loop phenomenon, so-called “repetition draw", occurs.

5.2 Preliminary and basic results

5.2.1 Model

We first model the game of open-hand BABANUKI. At first, let players be $\{P_0, P_1, \dots, P_{p-1}\}$ (p is an integer of 2 or more). We define the initial active player P_0 and the cyclic order of players as $\{P_0, P_1, \dots, P_{p-1}, P_0, \dots\}$. Let $[n] = \{1, 2, \dots, n\}$ and $[\bar{n}] = \{\bar{1}, \bar{2}, \dots, \bar{n}\}$ be the set of card faces. Card i and $\bar{i}$ forms a pair, and $\bar{i}$ is the complement of i, and i is the complement of $\bar{i}$. If a player has a pair of cards with a same number, he/she can discard them. For a set X of the cards, $|X|$ denotes the number of the cards in X.

In addition, there is a card called “Old maid", which has a special role. In this paper, we simply call it JOKER card because JOKER is often given the role in BABANUKI. This card has no complement and cannot be discarded from the hand. In the following, we model a pair of n number card and an one JOKER. The rule of open-hand BABANUKI game that we consider in this paper is as follows:

(Setting phase)

- All the cards are distributed to players, and if there is any pair of cards in each player hand, discard them. Let $i = 0$.

(Drawing phase)

- P_i selects an arbitrary card from P_{i-1}'s hand and adds that card to the hand, where the subscript of P_i is the modulo of p.
- If P_i has a same card with the drawn card, then discard the drawn card together with the same card. Otherwise, P_i adds the drawn card into his/her hand. In any case, this finishes the P_i's turn and $i := i + 1$.
- If all the cards are discarded, then the player leaves the game. If $p := 2$, the game ends. Otherwise, let $p := p - 1$, and edit the subscripts to fit the number of the remaining players, and go back to the beginning of Drawing phase.

(Ranking phase)

- The player who has discarded all his hands first ranks higher, and the player with the remaining hand until the end becomes the lowest.

5.2.2 2-player play

We give two basic results on 2-Player open-hand BABANUKI.

Theorem 5.1. *In 2-player Open-hand BABANUKI game, the player who does not have the JOKER card has a winning strategy.*

Theorem 5.2. *In 2-player Open-hand BABANUKI game, the player's best strategy for win is “draw any of the non-JOKER card in their turn".*

Zermelo proved that in 2-player perfect information combinatorial game, either of the players has a winning strategy or no player cannot win, which means a tie [50]. It is an existence theorem and does not provide a concrete winning strategy; it could be extremely hard to determine which player is the winner.

In these senses, non-trivial points of Theorems 1 and 2 are (1) if players are rational, no tie happens, and (2) determining the winner is very easy. Note that Zermelo's theorem is about 2-player case, and the cases of 3 or more players are very different; there are many examples that there exists no optimal strategy under natural preference assumptions [55]. In spite of that, we can show the existence of winning strategy of multiplayer open-hand BABANUKI , as we will see in Sections III and IV.

5.3 3-player play

In this section, we show that the winning strategy is determined by player's hands and order in 3-player open-hand Old Maid.

At first, we define some variables. In the following, it is assumed that the subscript i of P_i represents the modulo by 3, that is, i mod 3. X_i is the set of hands held by both players P_i and P_{i+1}. Similary, Y_i is the set of hands held by player P_i and player P_{i+2}, and Z_i is the set of hands held by both player P_{i+1} and player P_{i+2}. That is, $X_0 (= Y_1 = Z_2)$ is a set of hands that both P_0 and P_1 have, and similarly, $X_1 (= Y_2 = Z_0)$ is P_1 and P_2, $X_2 (= Y_0 = Z_1)$ is a set of hands that both P_2 and P_0 have (the definitions of Y_i and Z_i are redundant). Also, $j_i = 1$ when the player $P_i (i = 0, 1, 2)$ has JOKER, and $j_i = 0$ when the player does not have JOKER. Except for JOKER cards, there is no hand held by one player and one hand held by three players. Therefore, if the hand set of each player is $H_i (i = 0, 1, 2)$, $H_i = X_i \cup Y_i \cup \{j\}$ when P_i has JOKER card. When P_i does not have a JOKER card, $H_i = X_i \cup Y_i$.

The existence or non-existence of a player who has a winning strategy depends on the positional relationship between the active player and the player holding JOKER. The positional relationship between the active player and the JOKER player is the following three types:

(I) When the active player and the JOKER player are different, and the active player is next to the JOKER player.

(II) The active player and the JOKER player are different, and the active player is in the order of drawing the cards from the JOKER player.

(III) When the active player and the JOKER player are the same.

The winning condition in each case can be shown by the following theorem. First, the winning strategy holder in the situation (I) is characterized by the following theorem.

Theorem 5.3. *The active player is P_0, P_1 has* JOKER card. *At this time P_0 has a winning strategy if and only if:*

- $|X_0| = 1$, $|Y_0|$ *is odd,* $|Z_0| > 0$.
- $|X_0| = 0$, $|Y_0|$ *is even (and not 0),* $|Z_0| = 0$.
- $|X_0| = 0$, $|Z_0| > 0$.

P_1 has a winning strategy if and only if:

- $|X_0|$ *is odd,* $|Y_0| > 1$, $|Z_0| = 0$.

P_2 has a winning strategy if and only if:

- $|X_0| > 0$, $|Y_0| = 0$.
- $|Y_0| = 1$, $|Z_0| = 0$.
- $|X_0|$ *is even,* $|Y_0|$ *is odd number,* $|Z_0|$ *is even number,* $|X_0| \geq |Z_0|$.
- $|X_0|$ *is odd number,* $|Y_0|$ *is even number,* $|Z_0|$ *is odd number,* $|X_0| \geq |Z_0|$.

Note that if the condition of (I) does not meet the conditions of the theorem, no player has a winning strategy. This does not mean that no player holds the winning strategy throughout the game, but that the player holding the (future) winning strategy can change depending on each turn decision.

For example, if a player P_0 draws a certain card, P_1 will have a winning strategy in the next phase, but if another player draws, P_2 will have a winning strategy. Alternatively, no matter which card P_0 draws, the winning strategy holder may still not be determined at that point. From the above, it is necessary to show the theorems that characterize winning strategy holders not only in (I) but also in (II) and (III), separately from (I). The characterizations corresponding to (II) and (III) are as follows.

Theorem 5.4. *The active player is P_0, P_0 has* JOKER card. *At this time P_0 has a winning strategy if and only if:*

- $|X_0| = 0$, $|Y_0|$ *is odd number,* $|Z_0| > 0$.

P_1 has a winning strategy if and only if:

- $|Y_0| > 1$, $|Z_0| = 0$.
- $|X_0| = 1$, $|Z_0| = 1$.

- $|X_0|$, $|Y_0|$ *and* $|Z_0|$ *are odd number,* $|X_0| \leq |Y_0|$.
- $|X_0|, |Y_0|$ *and* $|Z_0|$ *are and even number,* $|X_0| \leq |Y_0|$.

P_2 *has a winning strategy if and only if:*

- $|X_0| > 0$, $|Y_0| = 0$.
- $|X_0| > 0$, $|Y_0| = 1$, $|Z_0|$ *is even number.*
- $|X_0| = 0$, $|Y_0| = 0$, $|Z_0|$ *is odd number.*

Theorem 5.5. *The active player is* P_0*,* P_2 *has* JOKER card. *At this time* P_0 *has a winning strategy if and only if:*

- $|X_0| = 0$, $|Z_0| > 0$.
- $|X_0| = 1$, $|Y_0| = 1$, $|Z_0| > 0$.
- $|X_0|$ *is odd number,* $|Y_0|$ *is odd number,* $|Z_0|$ *is even number,* $|Y_0| \leq |Z_0|$.
- $|X_0|$ *is even number,* $|Y_0|$ *is even number,* $|Z_0|$ *is odd number,* $|Y_0| \leq |Z_0|$.

P_1 *has a winning strategy if and only if:*

- $|Y_0| > 1$, $|Z_0| = 0$.
- $|X_0|$ *is even number,* $|Y_0| > 0$, $|Z_0| = 0$.
- $|X_0|$ *is odd number,* $|Y_0| > 1$, $|Z_0| = 1$.

P_2 *has a winning strategy if and only if:*

- $|X_0| > 0$, $|Y_0| = 0$, $|Z_0|$ *is even number.*

Note that a player that has no card in his/her hand at the beginning of the draw phase is obviously the winner and we omitted the case from the theorem.

5.4 4-player play

In this section, we show that there is a situation that all the each player's best strategy in BABANUKI game with hand-opening played by four players is a repetition-play

Theorem 5.6. *There is a situation where the situation is the repetition is the best strategy when aiming for all the players not to be the lowest in the open-hand BABANUKI played by four players.*

We show an example that repetition draw occurs.

Lemma 5.7. *Let the order of the players be P_1, P_2, P_3, P_4, and the next is P_0's turn. At this time, if each player takes the best action to improve his/her ranking, it occurs repetition draw in the case of the following hand arrangement.*

- *Players P_1 and P_3 have the same hand, and their number of hand is* 2*,*
- *The number of P_2's hand is* 2 *and the number of P_2's hand is* 3*.*

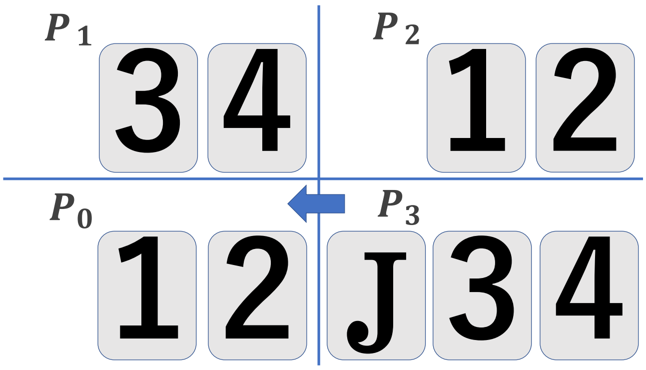


Figure 5.1: one of the same situation of Lemma 7

Chapter 6

SHICHINARABE

6.1 Introduction

SHICHINARABE is one of the most popular card games played in Japan. In this chapter, we consider SHICHINARABE, where the input consists of both a board and cards. We model a graphical generalization of SHICHINARABE and investigate the time complexity of the winner decision. Through the graphical generalization, we present a linear-time algorithm that can decide the winner of a given ordinary SHICHINARABE instance; the ordinary SHICHINARABE is shown to be an easy game. On the other hand, SHICHINARABE has many variations on how to use JOKER cards and with/without tunnel rules.

We formally define such rules in the graphical models, and we see the effect of the strength of rule sets. Concretely, we pick up trees and planar graphs as typical layouts of a board, and investigate the time complexity for natural combinations of graph classes and rule sets. As a result, the winner decision of graphical SHICHINARABE on trees is proved to be solvable in polynomial time, whereas that on planar graphs is shown to be hard to solve; the winner decision is NP-hard in general, and it is even PSPACE-hard if we adopt a generalized tunnel rule.

6.2 Preminaries

To introduce a graphical SHICHINARABE, we here define several notations on graphs used in this chapter. We assume basic knowledge about graph theory. Let $G = (V, E)$ be an directed or undirected graph $G = (V, E)$ where $n = |V|$ and $m = |E|$. We divide a set of vertex V into V_B, V_R, V_W. We call each of them a set of *blue vertex*, *red vertex* and *white vertex*.

For an undirected graph $G = (V, E)$, a vertex u is called a *neighbor* of v if there exists an edge $\{u, v\} \in E$. We denote by $N_G(v)$ the set of neighbors of v in

G, that is, $N_G(v) = \{u \in V \mid \{u, v\} \in E\}$.

The two players in the game will be referred to as *red player* and *blue player*. Under this setup, the game proceeds as follows.

The rules of 2-player graph SHICHINARABE

- We first determine which player is the first player, and they take turns in the order of first player, second player, first player, and so on.
- In a turn of each player, they select one of their vertices (the red vertices for the red player, the blue vertices for the blue player) neighboring a white vertex. The selected vertex changes to a white vertex. After the change, the turn action ends and moves to the next turn.
- If a player in a turn selects the last (unique) vertex of their color, the player wins. The game ends.
- If a player in a turn cannot select a vertex of their color in spite that a vertex of their color is left, that is, all the vertices of their color are not neighboring to a white vertex, the player loses. The game ends.

This is a generalization of the two-player SHICHINARABE game. Each vertex corresponds to a card, the red vertex to the red player's hand, the blue vertex to the blue player's hand, and the white vertex to the "7" card and the cards that players have already played.

For example, the ordinary 52 cards game playing corresponds to a graphical SHICHINARABE game playing on a spider graph where the length of every leg is 6, the number of legs is 8 and the central vertex is the unique white one at the beginning of the game.

If a player in a turn cannot select a vertex of their color in spite that a vertex of their color is left, the player loses in this game, which means the number of passes per player is limited to 0 while the number of passes per player is usually allowed to be around 3 in ordinary SHICHINARABE plays. This is also a natural extension of the following theorem.

Theorem 6.1. *For any p, the following two conditions are equivalent for a two-player SHICHINARABE game on a graph.*

- *In the SHICHINARABE game on graph G with 0 pass constraints, the red player has a winning strategy.*
- *In the SHICHINARABE game on graph G with p pass constraints, the red player has a winning strategy.*

We use the following theorem of Zermelo to prove this theorem.

Proposition 6.2. *In a combinatorial game of finite length with no draws, one of the players has a winning strategy [50].*

It should be noted that the graphical SHICHINARABE in this study satisfies all of the conditions for the application of Zermelo's theorem: it is a game without draws, it is a two-player game, it is a perfect information game, it is a sequential move game, it is finite game, and there is no chance factor affecting the actions of each player. To use this theorem, we show the proof of Theorem 6.1.

Proof. First, we show that sufficient condition holds.

[Proof of sufficient condition] In a game with 0 pass constraints, consider the phase G in which the left player can win. When playing this phase G in a constrained game with p passes, the left player can use the winning strategy in the constrained game with 0 pass constraints except for right player select a pass. If the right player select a pass, the left player can select a pass immediately after right player select a pass. In other words, whenever the right player selects a pass, the left player also selects a pass. In this way, the right player can win by the same procedure as in the constrained game with 0 passes.

Next, we show that necessary conditions.

[Proof of necessary conditions] To show necessary conditions, we show that "If G is a must-win game for the left player with p pass constrained game, then G is a must-win for the left player in a path 0 pass constrained game". This is a proposition P. Let this contrapositive be a proposition P'. Proposition P' is "If G is not a left-player must-win in with 0-pass constrain, then G is also not a left-player must-win with p-pass constrain". On the other hand, by Zermelo's theorem, all graphical SHICHINARABE games that are not left-player must-win are right-player must-win. Therefore, the proposition P' is equivalent to "If G' is right-player must-win in a path 0-constrained game, then it is right-player must-win in a path p-constrained game. This proposition P' can be shown to be true by an argument almost identical to that of sufficiency. Therefore, the proposition P is also true.

From the above discussion, this theorem is shown. □

These are the basic rules, and all variants in original SHICHINARABE are based on them. In addition to the above, SHICHINARABE is also known for its many localized rules. The typical local rules are (1) all mighty card, and (2) tunnel rule. We explain them in order.

(1)At first, we examine graph SHICHINARABE with all-mighty cards. All-mighty cards are one of the roles of JOKER cards in SHICHINARABE played with playing cards, and are played with several different rules depending on regions and communities. In this study, we define the All-mighty Right as "the right to change any vertex of the opponent's color (blue if the opponent's color is red, or red if the opponent's color is blue) to a white vertex before the turn action (followed by the turn action). In other words, we define "All-mighty Right" as "the right to play a card in place of a card you do not own at any time" in the ordinary SHICHINARABE game, and discuss its graphical generalized game. Note that this right can be exercised only the number of times permitted in advance, and not more than the number of times specified for each game. In some games, there is a rule that the JOKER card is given to the other player after the JOKER card is used, but we do not adopt this rule.

(2)We define edges with special vertex labels to correspond to the "tunnel rule." The tunnel rule stipulates that when the card of A(K) is placed on the board, the neighbor relation between A and K cards appears, while the ascending (descending) order neighbor relation of the suits disappears.

In this study, we define the following two edges with vertex labels to correspond to this tunnel rule:

Definition 6.3 (Edge Addition)**.** *$G = (V, E)$, which consist of the vertices set $V = V_B, V_R, V_W$ is the graph that represents the game board. We define the edge $e_A \in E$ as deletional edge of the vertex $\{v_A\} \in V$ as follows: If players change v to a white vertex, then the new graph G' obtained by adding e_A from G, consists of the new vertices set $V_B \setminus V_A, V_R \setminus V_A, V_W \cup v_A$ and new edge set $E \cup e_A$.*

Definition 6.4 (Edge deletion)**.** *$G = (V, E)$ is the graph that represents the game board. We define the edge $e_D \in E$ as deletional edge of the vertex $\{v_D\} \in V$ as follows: If players change v to a white vertex, then the new graph G' obtained by deleting e_D from G, consists of the new vertices set $V_B \setminus V_A, V_R \setminus V_A, V_W \cup v_A$ and new edge set $E \setminus e_D$.*

6.3 Time complexity

First, the following theorem holds for the amount of computation when the number of times the all-mighty right can be used is limited.

Theorem 6.5. *The decision of the winner of a graphical SHICHINARABE game on a tree can compute in $\mathcal{O}(n)$ even if the all-mighty right can be used at most once.*

Considering that the widely-played ordinary SHICHINARABE corresponds to the graphical SHICHINARABE on the spider graph, and that JOKER cards correspond to the all-mighty right, the ordinary SHICHINARABE, which contains at most one JOKER card, can be computed in linear time of the total number of cards. However, in the SHICHINARABE game, there are some local rules in which multiple JOKER cards are used or JOKER cards are moved for each use. The following theorem holds for the computational complexity when there are more than one JOKER cards in such an ordinary SHICHINARABE.

Theorem 6.6. *The determination of the winner of a graphical SHICHINARABE game on a tree, where only one of the players has the right to use the all-mighty right k times, can compute in $\mathcal{O}(n^k)$ time.*

Theorem 6.7. *The determination of the winner of a graphical SHICHINARABE game on a tree, where only one of the players has the right to use the all-mighty right multiple times, can compute in $\mathcal{O}(n^4)$ time.*

These results show that it can be computed in $\mathcal{O}(n^k)$ time if there are three or fewer all-mighty rights, and in $\mathcal{O}(n^4)$ time if there are four or more all-mighty rights.

So far, we have only considered games on tree-graphs or their subclass called spider graph, but given that the game is played on a board, it is natural to consider a generalization of this graph game to planar graphs. For graphical SHICHINARABE games on planar graphs, the following theorems hold.

Theorem 6.8. *It is NP-hard to determine which of the two players is the winner in a graphical SHICHINARABE on a planar graph where only one of the players has the right to use the all-mighty right multiple times.*

Theorem 6.9. *It is PSPACE-hard to determine which of the two players is the winner in a graphical SHICHINARABE on a planar graph where the Edge deletion rule and Edge addition rule are adapted.*

Chapter 7

Conclusion

In this thesis, we investigate the time complexity of winner decisions in card-discarding games, which are widely played in the world.

In Chapter 2 and Chapter 3, we showed that it can be decided in $\mathcal{O}(n)$ time which player has a winning strategy in 2-player TANHINMIN and TANHINMIN with cut cards. It should be noted that the decision can be done at any moment of the match. In Chapter 4, we modeled TANHINMIN with structural oracles to identify the essential information to construct a winning strategy. The oracle model that we propose in this thesis can qualify and quantify the information that each player can receive during plays. The obtained results show that in order to play the winning strategy the full information of the game is not necessarily needed. Figure 4.1 summarizes the power of oracles and solvability of the imperfect variant of TANHINMIN.

In Chapter 5, we investigated the existence of a winning strategy on open-hand BABANUKI. Although the 2-player case is almost trivial, 3 or more players' cases are not. In fact, for 3-player case, there are initial hands where no player has a winning strategy, though we give an explicit necessary and sufficient condition that a player has a winning strategy. The 4-player case is more complicated, and there is an example that the best strategy of every player is to draw JOKER, which yields an infinite repetition play.

In Chapter 6, we study SHICHINARABE which is generalized to a graphical setting, called Graph SHICHINARABE. For the graphical SHICHINARABE on trees includes an ordinary SHICHINARABE, we can compute the winner in linear time, whereas it is harder to compute the winner in the SHICHINARABE with all-mighty card or tunnel rule. In fact, it is NP-hard to compute the winner in the graphical SHICHINARABE on planer graphs with all-mighty cards. Furthermore, it is PSPACE-hard to compute the winner in the graphical SHICHINARABE on planer graphs with tunnel rule.

Before concluding this thesis, we give indications about further research directions of each topic. As explained in Chapter 1, TANHINMIN is a very simplified variant of DAIHINMIN. The major differences from DAIHINMIN are that it is an imperfect information game, and that several interesting and important rules are not adopted. Since a purpose of this research is to investigate the mathematical nature of DAIHINMIN via TANHINMIN, it is natural to extend the research to focus on not the former but the latter; investigating TANHINMIN with extra rules. Among many rules, the following are most popular ones: discarding multiple cards with the same strength (we call *multi-cards rule*), revolution. Discarding multiple cards means that if the table is empty, then the active player can discard multiple cards with the same strength, such as three "4". Then the next player must discards three stronger cards with the same strength, such as three "5". This is maybe the most important rule of DAIHINMIN that TANHINMIN is not equipped with. Revolution is a rule to change the order of strength by the active player discarding some special card(s) at the table. Typically, if the active player discards some four cards with the same strength (e.g., four "5") and no other player can discard any cards, then the table is set empty and from the next turn the ordering of strength is reversed; 1 becomes the strongest and n becomes the weakest. There are many variations of revolution rule. All of these rules are interesting and important, but it might be difficult to extend the analysis to TANHINMIN with multi-cards rule or TANHINMIN with revolution rule. To analyze these rules is natural future work. It should be noticed that this oracle-based analysis framework proposed in this thesis has several benefits. Applying the framework to some other games would be interesting.

There are also several open questions in BABANUKI game. One is to investigate whether the k-player case for every $k \geq 5$ also has an example of an infinite repetition play like the 4-player case. It might be difficult to prove because we need to show that the best play of each player is a repetition draw, but which is best is not easily verified.

In the graphical SHICHINARABE in trees including general SHICHINARABE, we can compute the winner in linear time, but it might be difficult to design a linear-time algorithm for the winner decision in the SHICHINARABE on trees with all-mighty card or tunnel rules. As we see, it is NP-hard to compute winner in the graphical SHICHINARABE in planar graphs with all-mighty cards, but our hardness proof utilizes the power of all-mighty cards. The complexity of deciding the winner of the graphical SHICHINARABE in planar graphs without an all-mighty card remains open.

As mentioned in Chapter 1, the card discarding game, which deals with board information, is widely played, although few studies have been conducted on its computational time complexity. In this study, we found that the computational difficulty of the game is comparable to that of other board games such as planar graphs. In other words, card discarding games with board information have the same computational complexity as other games with board information. Of course, this study on the computational time complexity winner of card discarding games with board information is only one example, but we hope that this research will be useful for future studies as a starting point for characterizing “difficult” games.

Bibliography

[1] Pagat.com. `https://www.pagat.com/`. (Accessed on 01/12/2021).

[2] The sveriges riksbank prize in economic sciences in memory of alfred nobel. `https://www.nobelprize.org/prizes/economic-sciences/`. (Accessed on 01/11/2021).

[3] Wikipedia. `https://en.wikipedia.org/wiki/Daifugo`, (Accessed 2021-01-11).

[4] ABUKU, T., FUKUI, M., SAKAI, K., AND SUETSUGU, K. On a combination of the cyclic nimhoff and subtraction games. *Tsukuba Journal of Mathematics 43*, 2 (2019), 241–249.

[5] ALBERT, M. H., NOWAKOWSKI, R. J., AND WOLFE, D. *Lessons in play: an introduction to combinatorial game theory*. CRC Press, 2019.

[6] ALLIS, L. V. A knowledge-based approach of connect four: The game is over, white to move wins. *report No. IR-163* (1988).

[7] ALLIS, L. V., ET AL. *Searching for solutions in games and artificial intelligence*. Ponsen & Looijen Wageningen, 1994.

[8] ALLIS, L. V., VAN DEN HERIK, 1, H., AND HUNTJENS, M. P. Go-moku solved by new search techniques. *Computational Intelligence 12*, 1 (1996), 7–23.

[9] BERLEKAMP, E. R., CONWAY, J. H., AND GUY, R. K. *Winning Ways for Your Mathematical Play*. A K Peters/CRC Press, 2001.

[10] BLAIR, J. R., MUTCHLER, D., AND LIU, C. Games with imperfect information. In *Proceedings of the AAAI Fall Symposium on Games: Planning and Learning, AAAI Press Technical Report FS93-02, Menlo Park CA* (1993), pp. 59–67.

[11] BODLAENDER, H. L., KRATSCH, D., AND TIMMER, S. T. Exact algorithms for kayles. *Theoretical Computer Science 562* (2015), 165–176.

[12] Bouton, C. L. Nim, a game with a complete mathematical theory. *The Annals of Mathematics 3*, 1/4 (1901), 35–39.

[13] Bowling, M., Burch, N., Johanson, M., and Tammelin, O. Heads-up limit hold'em poker is solved. *Science 347*, 6218 (2015), 145–149.

[14] Brown, N., and Sandholm, T. Superhuman ai for multiplayer poker. *Science 365*, 6456 (2019), 885–890.

[15] Carvalho, A., Neto, J. P., and dos Santos, C. P. Combinatorics of jenga. *AUSTRALASIAN JOURNAL OF COMBINATORICS 76*, 1 (2020), 87–104.

[16] Dailly, A., Gledel, V., and Heinrich, M. A generalization of arc-kayles. *International Journal of Game Theory 48*, 2 (2019), 491–511.

[17] Demaine, E. D., Demaine, M. L., Uehara, R., Uno, T., and Uno, Y. Uno is hard, even for a single player. In *International Conference on Fun with Algorithms* (2010), Springer, pp. 133–144.

[18] Elkies, N. D. On numbers and endgames: combinatorial game theory in chess endgames. *Games of No Chance 29* (1996), 135–150.

[19] Fraenkel, A. Combinatorial games: selected bibliography with a succinct gourmet introduction. *The Electronic Journal of Combinatorics* (2012), DS2–Aug.

[20] Fraenkel, A. S., and Goldschmidt, E. Pspace-hardness of some combinatorial games. *Journal of Combinatorial Theory, Series A 46*, 1 (1987), 21–38.

[21] Furtak, T., Kiyomi, M., Uno, T., and Buro, M. Generalized amazons is pspace-complete. In *IJCAI* (2005), Citeseer, pp. 132–137.

[22] Gale, D. A curious nim-type game. *The American Mathematical Monthly 81*, 8 (1974), 876–879.

[23] Grundy, P. M. Mathematics and games. *Eureka 2* (1939), 6–9.

[24] Guy, R. K., and Smith, C. A. The g-values of various games. In *Mathematical Proceedings of the Cambridge Philosophical Society* (1956), vol. 52, Cambridge University Press, pp. 514–526.

[25] Hearn, R. A. Amazons, konane, and cross purposes are pspace-complete. In *Games of No Chance III, Proc. BIRS Workshop on Combinatorial Games* (2005), pp. 287–306.

[26] HEARN, R. A., AND DEMAINE, E. D. *Games, Puzzles, and Computation*. A K Peters, 2009.

[27] HEARN, R. A., AND DEMAINE, E. D. *Games, puzzles, and computation*. CRC Press, 2009.

[28] HOPCROFT, J. E., AND KARP, R. M. An n^5/2 algorithm for maximum matchings in bipartite graphs. *SIAM Journal on computing 2*, 4 (1973), 225–231.

[29] ITO, H., NAGAO, A., AND PARK, T. Generalized shogi, chess, and xiangqi are constant-time testable. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences 102*, 9 (2019), 1126–1133.

[30] IWATA, S., AND KASAI, T. Shogi on n x n board is complete in exponential time (in japanese). *IEICE TRANSACTIONS 70*, 10 (1987), 1843–1852.

[31] IWATA, S., AND KASAI, T. The othello game on an n$\times$ n board is pspace-complete. *Theoretical Computer Science 123*, 2 (1994), 329–340.

[32] KRAWEC, W. O. Analyzing n-player impartial games. *International Journal of Game Theory 41*, 2 (2012), 345–367.

[33] KRAWEC, W. O. n-player impartial combinatorial games with random players. *Theoretical Computer Science 569* (2015), 1–12.

[34] LIU, W. A., AND DUAN, J. W. Misère nim with multi-player. *Discrete Applied Mathematics 219* (2017), 40–50.

[35] LIU, W. A., AND WANG, M. Y. Multi-player subtraction games. *Theoretical Computer Science 659* (2017), 14–35.

[36] LIU, W. A., AND ZHOU, J. J. Multi-player small nim with passes. *Discrete Applied Mathematics 236* (2018), 306–314.

[37] MOORE, E. H. A generalization of the game called nim. *The Annals of Mathematics 11*, 3 (1910), 93–94.

[38] MYERSON, R. B. *Game theory*. Harvard university press, 2013.

[39] NAKAMURA, T. Counting liberties in go capturing races. In *Games of No Chance III, Proc. BIRS Workshop on Combinatorial Games* (2005), pp. 177–196.

[40] NISHINO, J. An analysis on tanhinmin game (in japanese). *IPSJ Symposium Series 2007*, 12 (2007), 66–73.

[41] NISHINO, J., AND NISHINO, T. Parallel monte carlo search for imperfect information game daihinmin. In *Parallel Architectures, Algorithms and Programming (PAAP), 2012 Fifth International Symposium on* (2012), IEEE, pp. 3–6.

[42] OKUBO, S., AYABE, T., AND NISHINO, T. Cluster analysis using n-gram statistics for daihinmin programs and performance evaluations. *International Journal of Software Innovation (IJSI) 4*, 2 (2016), 33–57.

[43] PATASHNIK, O. Qubic: $4\times 4\times 4$ tic-tac-toe. *Mathematics Magazine 53*, 4 (1980), 202–216.

[44] ROBSON, J. Combinatorial games with exponential space complete decision problems. In *International Symposium on Mathematical Foundations of Computer Science* (1984), Springer, pp. 498–506.

[45] ROBSON, J. M. The complexity of go. In *Proc. 9th World Computer Congress on Information Processing, 1983* (1983), pp. 413–417.

[46] ROBSON, J. M. N by n checkers is exptime complete. *SIAM Journal on Computing 13*, 2 (1984), 252–267.

[47] ROUGHGARDEN, T. Algorithmic game theory. *Communications of the ACM 53*, 7 (2010), 78–86.

[48] SCHAEFER, T. J. On the complexity of some two-person perfect-information games. *Journal of Computer and System Sciences 16*, 2 (1978), 185–225.

[49] SCHAEFFER, J., BURCH, N., BJÖRNSSON, Y., KISHIMOTO, A., MÜLLER, M., LAKE, R., LU, P., AND SUTPHEN, S. Checkers is solved. *science 317*, 5844 (2007), 1518–1522.

[50] SCHWALBE, U., AND WALKER, P. Zermelo and the early history of game theory. *Games and economic behavior 34*, 1 (2001), 123–137.

[51] SIEGEL, A. N. *Combinatorial Game Theory*. American Mathematical Society, 2013.

[52] SILVER, D., HUBERT, T., SCHRITTWIESER, J., ANTONOGLOU, I., LAI, M., GUEZ, A., LANCTOT, M., SIFRE, L., KUMARAN, D., GRAEPEL, T., ET AL. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science 362*, 6419 (2018), 1140–1144.

[53] SILVER, D., SCHRITTWIESER, J., SIMONYAN, K., ANTONOGLOU, I., HUANG, A., GUEZ, A., HUBERT, T., BAKER, L., LAI, M., BOLTON, A., ET AL. Mastering the game of go without human knowledge. *nature 550*, 7676 (2017), 354–359.

[54] SPRAGUE, R. Über mathematische kampfspiele, tohuky math. *J 41* (1935), 438–444.

[55] SUETSUGU, K. Multiplayer games as extension of misère games. *International Journal of Game Theory 48*, 3 (2019), 781–796.

[56] SUTTLE, J. P., AND JONES, D. A. Poker game, June 6 1989. US Patent 4,836,553.

[57] TANAKA, S., BONNET, F., TIXEUIL, S., AND TAMURA, Y. Quixo is solved. *arXiv preprint arXiv:2007.15895* (2020).