Vol.2020-MPS-131 No.2
2020/12/17

Modeling Imperfect Information TANHINMIN with Structural Oracle

Hironori Kiya[1,a)] Katsuki Ohto[,b)] Hirotaka Ono[1,c)]

Abstract: TANHINMIN is a simplified and perfect information variant of DAIHINMIN game, which is major playing card game in Japan. It is known that it can be decided in linear time which player has a winning strategy in 2-player TANHINMIN game. This paper is concerned 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.

1. Introduction

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 [2]; 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 [5, 6]. 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 the game AI programs are getting stronger every year [3, 4], the mathematical nature of the game itself is still mostly unknown. DAIHINMIN games contain various special rules 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 consumption-type 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 a set of cards, 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 can discard any card, then the turn ends and the player who last discarded a set of cards can start a new turn by discarding any set of cards. After several turns, the first player that has discarded all her cards is the winner. The basic rule of TANHINMIN is the same as DAIHINMIN, but it is very 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. For a more detailed explanation, see Section 2, where the formal definition and a concrete play example of TANHINMIN game are given.

Since this 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. Although this does not immediately imply that the winner decision is easy, we can decide the winner of a given 2-player perfect information variant of TANHINMIN in linear time [1]. 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 this paper, we model TANHINMIN with structural oracles to identify the essential information to construct a

1 Department of Mathematical Informatics,
Nagoya University, Nagoya 464-8601, Japan.
a) kiya.hironori@gmail.com
b) katsuki.ohto@gmail.com
c) ono@i.nagoya-u.ac.jp

winning strategy. As will be described later in section 2, 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 paper 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 [1]. 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. 2.

The rest of this paper is organized as follows: In Section 2, we introduce the rules of TANHINMIN, and also a graph model for analysis. In Section 3, we introduce our oracle models and summarize the main contribution, where the proofs are omitted. Finally, Section 4 concludes the paper with some further remarks.

2. TANHINMIN Rules and Notations

2.1 The rule of TANHINMIN

We first model a game of TANHINMIN. Let $[n] = \{1, 2, \dots, n\}$ 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, ... 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). Each player in her turn discards 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, she selects "pass". Then let the table be empty and go to the next turn. The player that first discards all the cards in her hand is the winner.

This is the basic rule of TANHINMIN. To investigate a winning strategy of 2-player TANHINMIN, we impose the following additional rule: if a player selects pass, then the next player cannot select pass. This is because two or more consecutive passes are useless, though we omit a formal proof of this in this paper.

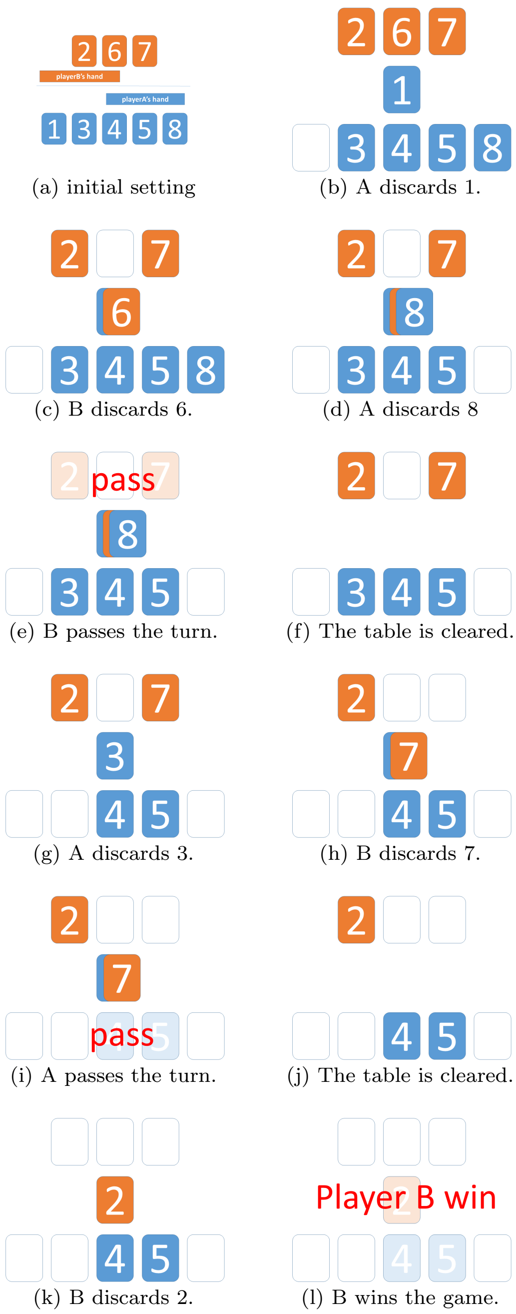

Fig. 1 A play example of 2-player TANHINMIN

In Figure 1, we show a play example of 2-player TANHINMIN game. 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 selects 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 her card, player B is the winner (Fig. 1 (l)).

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 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 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 1. ([1]) Given a configuration (X_0, X_1, r) of 2-player TANHINMIN with n cards, P_0 has a winning strategy when $\mu_0 > \mu_1$ holds, 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.

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 [1].

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.

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

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

It is interesting that Theorems 2 and 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. 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 $\mu_0 > \mu_1$ ($\mu_0 \le \mu_1$) at the timing.

Theorems 3 and 4 contrast well. Theorem 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 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., $\mu_0 \le \mu_1$) of timing t changes $\mu_0 \le \mu_1$ (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 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 5 and Corollary 6 may contrast. Theorem 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 6 implies that if a player can access the matching size oracle every time, she can adjust her strategy to the winning one.

Figure 2 summarizes the results.

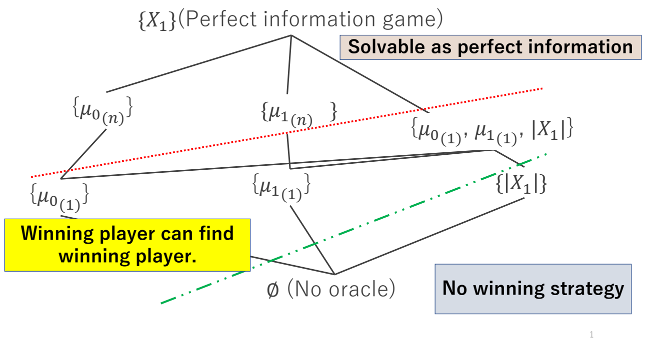

Fig. 2 Relationship between accessible oracles and solvability of the winner decision

In Figure 2, 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. Concluding remarks

In this paper, we modeled TANHINMIN with structural oracles to identify the essential information to construct a winning strategy. The oracle model that we propose in this paper 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 2 summarizes the power of oracles and solvability of the imperfect variant of TANHINMIN.

It should be noticed that this oracle-based analysis framework proposed in this paper has several benefits. Applying the framework to some other games would be interesting future work.

References

[1] Kiya, H., Ohto, K. and Hirotaka, O.: Deciding the winning player in 2-player 8-cut rule TANHINMIN game (in Japanese), *IPSJ Symposium Series*, Vol. 2018, No. 3, pp. 1–5 (2018).

[2] Nishino, J.: An analysis on TANHINMIN game (in Japanese), *IPSJ Symposium Series*, Vol. 2007, No. 12, pp. 66–73 (2007).

[3] Nishino, J. and Nishino, T.: Parallel Monte Carlo search for imperfect information game Daihinmin, *Parallel Architectures, Algorithms and Programming (PAAP), 2012 Fifth International Symposium on*, IEEE, pp. 3–6 (2012).

[4] 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)*, Vol. 4, No. 2, pp. 33–57 (2016).

[5] Pagat.com: https://www.pagat.com, (accessed 2020-11-10).

[6] Wikipedia: https://en.wikipedia.org/wiki/Daifugo, (accessed 2020-11-10).