An Application of Secure Computation to Tagiron[⋆]

Koichi Koizumi[1] and Takaaki Mizuki[2]

[1] National Institute of Technology, Fukushima College, Iwaki, Japan
[2] Cyberscience Center, Tohoku University, Sendai, Japan

Abstract. Tagiron (also known as Break the Code) is a popular logic-based deduction game played with a deck of 20 cards. The rules of Tagiron vary depending on the number of players. In a four-player game, each player is dealt four cards, and the remaining four cards, called the target cards, are the goal for players to deduce. When a player makes a "challenge" to deduce the target cards, they secretly look at the hidden cards directly. If a player makes an incorrect challenge, they immediately lose and are eliminated from the game as they gain complete knowledge of the target cards. For this reason, players are limited to a single challenge throughout a game. In this paper, we propose a secure computation protocol using card-based cryptography. This protocol can determine the success or failure of a challenge without revealing the hidden target cards to any player. This innovation opens up new variations of Tagiron that allow for multiple challenges (e.g., up to two challenges), which could potentially enhance gameplay and strategic depth.

Keywords: Card-based cryptography · Secure computation · Playing cards · Tagiron

1 Introduction

This paper discusses *Tagiron* (also known as *Break the Code*[3]), a popular logic-based deduction game invented by Ryohei Kurahashi. The game is played by two to four players using a deck of 20 *Tagiron cards* and a deck of 21 *question cards*. Since it was first published in 2014, several game versions have been released by various international publishers; Fig. 1 shows one of the Japanese versions.

While the objective of the game varies depending on the number of players, this paper focuses on the four-player version and begins by explaining its rules.

1.1 Rules of Tagiron

The game uses the 20 Tagiron cards (small tiles) shown in Fig. 2. These cards are marked with numbers from 0 to 9 and have a color of red, blue, or green; we denote them by

[⋆] This paper appears in Proceedings of ACG 2025. This version of the contribution has been accepted for publication, after peer review but is not the Version of Record and does not reflect post-acceptance improvements, or any corrections. The Version of Record is available online at: The Version of Record will be available online soon. Use of this Accepted Version is subject to the publisher ' s Accepted Manuscript terms of use: https://www.springernature.com/gp/open-research/policies/accepted-manuscript-terms.

[3] https://iellogames.com/games/break-the-code/

Fig. 1: Tagiron by JELLY JELLY GAMES (https://jelly2games.com/tagiron)

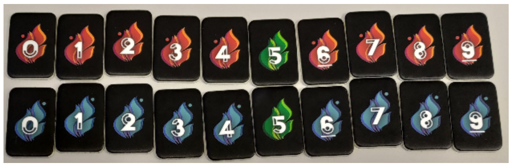

Fig. 2: Tagiron cards (tiles)

preceding the number with their color (R, B, or G), as in

$$\boxed{\text{R0}}\boxed{\text{B0}}\boxed{\text{R1}}\boxed{\text{B1}}\boxed{\text{R2}}\boxed{\text{B2}}\boxed{\text{R3}}\boxed{\text{B3}}\boxed{\text{R4}}\boxed{\text{B4}}\boxed{\text{G5}}\boxed{\text{G5}}\boxed{\text{R6}}\boxed{\text{B6}}\boxed{\text{R7}}\boxed{\text{B7}}\boxed{\text{R8}}\boxed{\text{B8}}\boxed{\text{R9}}\boxed{\text{B9}}.$$

Note that every card except for the two $\boxed{\text{G5}}$ cards is unique.

These 20 cards are shuffled and dealt to four players (denoted by P_1, P_2, P_3, P_4), as in a usual card game, such that each player receives four cards; thus, four face-down cards remain and become the *target cards*, which are the goal for the players to deduce. Each player arranges their four cards in ascending order from left to right, without revealing them, where the red card is smaller for any pair of red and blue cards with the same number. The following is an example of a placement of cards:

$$P_1 : \boxed{\text{R1}}\boxed{\text{B3}}\boxed{\text{B6}}\boxed{\text{R8}},\ P_2 : \boxed{\text{R4}}\boxed{\text{B4}}\boxed{\text{G5}}\boxed{\text{R7}},\ P_3 : \boxed{\text{B0}}\boxed{\text{B2}}\boxed{\text{R6}}\boxed{\text{B8}},\ P_4 : \boxed{\text{R0}}\boxed{\text{R3}}\boxed{\text{B7}}\boxed{\text{R9}},$$

where these cards are placed in a hidden form (e.g., they are placed face down) and the remaining four cards, namely the target cards, $\boxed{\text{G5}}\boxed{\text{B1}}\boxed{\text{B9}}\boxed{\text{R2}}$ are placed with keeping their faces down.

In addition to the 20 Tagiron cards, we use 19 question cards.[4] After shuffling the question cards, six of them are revealed and placed on the table, while the rest form a deck.

Players take turns performing the following actions:

1. A player chooses one of the six revealed question cards and asks a question.
2. All players, including the questioner, answer the question honestly.
3. The used question card is discarded, and a new one is revealed from the deck.

[4] Among the 21 question cards, only 19 are used, because the remaining two questions (which assume an odd number of player's cards) make no sense in the four-player version of Tagiron.

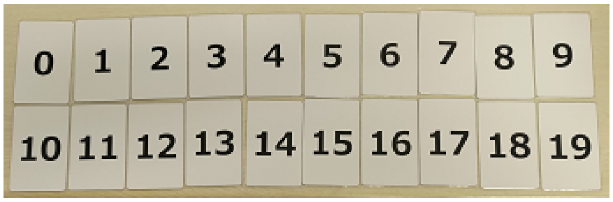

Fig. 3: Numbered cards for protocols

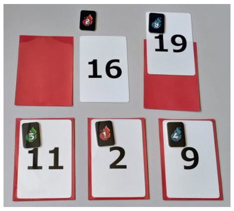

Fig. 4: Fixing two types of cards by sleeves

In the above example, if P_1 asks "Where is 9?", then P_1, P_2, and P_3 answer "None," and P_4 responds, "It is the rightmost one." From this, all players can deduce that there is exactly one '9' among the target cards. P_4, knowing that their hand contains $\boxed{\text{R9}}$, can further deduce that $\boxed{\text{B9}}$ must be one of the target cards. Other question cards include "How many blue numbers are there in total?", "How many odd numbers are there?", "What is the total sum of the smallest three numbers?", and so on.

Each player can make a *challenge* anytime, but only once during a game. The player who challenges writes down their guess for the numbers and colors of the four target cards. The player then looks at the four face-down target cards secretly. If their guess is correct, the player declares "Success" and wins the game. If their guess is incorrect, the player declares "Fail" and is eliminated. Eliminated players continue to answer questions but can no longer make challenges or ask questions. If no player can make a challenge, the game ends (although the termination condition can vary depending on the situation).

1.2 Tagiron's Challenge

As explained, in Tagiron, a player making a challenge directly examines the face-down target cards to see if their guess is correct; this process has a significant consequence: if the player's guess is wrong, they are eliminated from the game because they gain complete knowledge of the target cards.

This paper explores an alternative challenge mechanism. By using a *secure computation protocol*, we can determine the success or failure of a challenge without revealing the target cards to the player. Under this new challenge mechanism, a player who fails a challenge would only learn that their guess was incorrect, without gaining any further information. This allows the game to continue with all players still in play.

1.3 Contribution of This Paper

To provide a method for determining a challenge's outcome without directly viewing the target cards, this paper constructs *card-based protocols* that use a deck of cards to perform cryptographic functionalities (for a survey of card-based cryptography, see [8]).

Since the 20 Tagiron cards are very small tiles as illustrated in Fig. 2, we use regular-sized numbered cards, as shown in Fig. 3, to construct card-based protocols. That is, we use 20 cards $\boxed{0}\,\boxed{1}\cdots\boxed{19}$ numbered from 0 to 19, where the back of every card is $\boxed{?}$. We correspond them to the Tagiron cards, as follows:

$$\begin{array}{cccccccccccccccccccc}
\boxed{\mathrm{R0}} & \boxed{\mathrm{B0}} & \boxed{\mathrm{R1}} & \boxed{\mathrm{B1}} & \boxed{\mathrm{R2}} & \boxed{\mathrm{B2}} & \boxed{\mathrm{R3}} & \boxed{\mathrm{B3}} & \boxed{\mathrm{R4}} & \boxed{\mathrm{B4}} & \boxed{\mathrm{G5}} & \boxed{\mathrm{G5}} & \boxed{\mathrm{R6}} & \boxed{\mathrm{B6}} & \boxed{\mathrm{R7}} & \boxed{\mathrm{B7}} & \boxed{\mathrm{R8}} & \boxed{\mathrm{B8}} & \boxed{\mathrm{R9}} & \boxed{\mathrm{B9}} \\
\updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow \\
\boxed{0} & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} & \boxed{5} & \boxed{6} & \boxed{7} & \boxed{8} & \boxed{9} & \boxed{10} & \boxed{11} & \boxed{12} & \boxed{13} & \boxed{14} & \boxed{15} & \boxed{16} & \boxed{17} & \boxed{18} & \boxed{19}
\end{array}. \tag{1}$$

In our method, when the 20 Tagiron cards are dealt to players at the beginning of a game, the 20 numbered cards are also dealt together, such that the correspondence (1) is kept. To implement this, it suffices to place a Tagiron card and its corresponding numbered card into a single sleeve, as shown in Fig. 4, and shuffle the 20 sleeves.

After the two types of cards are dealt, the numbered cards of the players are collected together with the target numbered cards (while keeping their faces down):

$$\begin{array}{l}
P_1 : \boxed{?}\boxed{?}\boxed{?}\boxed{?},\ P_2 : \boxed{?}\boxed{?}\boxed{?}\boxed{?},\ P_3 : \boxed{?}\boxed{?}\boxed{?}\boxed{?},\ P_4 : \boxed{?}\boxed{?}\boxed{?}\boxed{?}, \\
\text{target cards} : \boxed{?}\boxed{?}\boxed{?}\boxed{?},
\end{array}$$

whereas the Tagiron cards are used as in the normal game.

Using the numbered cards along with a set of black cards $\boxed{\clubsuit}\boxed{\clubsuit}\cdots$ and a set of red cards $\boxed{\heartsuit}\boxed{\heartsuit}\cdots$, we will propose a method for securely determining whether a challenge succeeds or not without leaking any information (more than necessary). Our proposed method consists of two card-based protocols: the "preprocessing protocol" and the "challenge protocol." After the preprocessing protocol is executed once at the beginning, the challenge protocol is executed each time a player attempts a challenge. When making a challenge, the player secretly enters their challenge by placing a sequence of black and red cards face down. The protocol then secretly computes whether this input corresponds to the target cards.

Our method would open the door to new game variations of Tagiron. For example, a version could allow players up to two challenges, creating new strategic possibilities where a failed first challenge does not mean elimination.

1.4 Related Work

The history of card-based cryptography began in 1989 [1], and its development is now remarkable (e.g., [12–14]). To the best of our knowledge, the first 'practical' attempt to apply card-based cryptography to card games was made by Shinagawa, Miyahara, and Mizuki [21], who designed card-based protocols for creating virtual players in Old Maid. Subsequently, Ruangwises and Shinagawa [20] extended this work by constructing card-based protocols for UNO. In addition, card-based cryptography was used to create a new game, Gakmoro [15], and to enhance the Hit and Blow game [5].

Many card-based zero-knowledge proof protocols have been developed for various puzzles and games, such as the 15 puzzle [22], ABC end view [3], Dosun-Fuwari [7], Kurodoko [18], Moon-or-Sun [4], pancake sorting [10], Sudoku [17,23], Topswops [9], Usowan [11], and Zeiger [19].

2 Preliminaries

In this section, we describe some notations and existing protocols on card-based cryptography that are necessary to present our method.

2.1 Cards and Binary Encoding

The cards used in this paper are, as already seen in Section 1, black cards $\boxed{\clubsuit}\boxed{\clubsuit}\cdots$, red cards $\boxed{\heartsuit}\boxed{\heartsuit}\cdots$, and numbered cards $\boxed{0}\boxed{1}\ \cdots\boxed{19}$. Cards with the same face are assumed to be indistinguishable from each other, and all cards share a common back pattern $\boxed{?}$.

We follow the standard convention to represent a Boolean value by the order of a black card $\boxed{\clubsuit}$ and a red card $\boxed{\heartsuit}$:

$$\boxed{\clubsuit}\boxed{\heartsuit} = 0, \qquad \boxed{\heartsuit}\boxed{\clubsuit} = 1. \tag{2}$$

According to the encoding rule (2), when a bit $x \in \{0, 1\}$ is represented using two face-down cards, they are called a *commitment* to x. We sometimes place two cards constituting a commitment vertically, following the encoding:

$$\begin{array}{c}\boxed{\clubsuit}\\ \boxed{\heartsuit}\end{array} = 0, \qquad \begin{array}{c}\boxed{\heartsuit}\\ \boxed{\clubsuit}\end{array} = 1.$$

In the sequel, we use the following representations for describing a commitment to x:

$$\underbrace{\boxed{?}\boxed{?}}_{x} \qquad \underbrace{\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}}_{x} \qquad \left.\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}\right\}x \qquad x : \boxed{?}\boxed{?}.$$

2.2 Pile-Scramble Shuffle

The shuffling action used in this paper is the *pile-scramble-shuffle* [6], which randomly permutes a sequence of piles, each consisting of the same number of cards, without changing the order of the cards inside each pile.

For example, when applying a pile-scramble shuffle to four 3-card piles, denoted by p_1, p_2, p_3, p_4, we write the transition as:

$$\begin{array}{cccc}\boxed{?}&\boxed{?}&\boxed{?}&\boxed{?}\\ \boxed{?}&\boxed{?}&\boxed{?}&\boxed{?}\\ \boxed{?}&\boxed{?}&\boxed{?}&\boxed{?}\\ p_1&p_2&p_3&p_4\end{array} \rightarrow \left[\begin{array}{c||c||c||c}\boxed{?}&\boxed{?}&\boxed{?}&\boxed{?}\\ \boxed{?}&\boxed{?}&\boxed{?}&\boxed{?}\\ \boxed{?}&\boxed{?}&\boxed{?}&\boxed{?}\end{array}\right] \rightarrow \begin{array}{cccc}\boxed{?}&\boxed{?}&\boxed{?}&\boxed{?}\\ \boxed{?}&\boxed{?}&\boxed{?}&\boxed{?}\\ \boxed{?}&\boxed{?}&\boxed{?}&\boxed{?}\\ p_{i_1}&p_{i_2}&p_{i_3}&p_{i_4}\end{array},$$

where the outcome is uniformly randomly chosen from the 4! possibilities (and nobody knows which outcome occurs), i.e., $i_j = \pi^{-1}(j)$ for every j, $1 \leq j \leq 4$, such that π is a uniformly distributed random permutation chosen from the symmetric group of degree 4.

2.3 Copy Protocol

The existing copy protocol [16] takes a commitment $x \in \{0,1\}$ as input and outputs two identical commitments to x using four free cards (without leaking any information about x):

$$\underbrace{\boxed{?}\boxed{?}}_{x}\boxed{\clubsuit}\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\heartsuit} \Rightarrow \underbrace{\boxed{?}\boxed{?}}_{x}\underbrace{\boxed{?}\boxed{?}}_{x}\boxed{\clubsuit}\boxed{\heartsuit}.$$

Specifically, the procedure is as follows.

1. Place the six cards as follows, turn over the face-up cards, and apply a pile-scramble shuffle:

$$\begin{array}{rcc} x: & \boxed{?} & \boxed{?} \\ & \boxed{\clubsuit} & \boxed{\heartsuit} \\ & \boxed{\clubsuit} & \boxed{\heartsuit} \end{array} \rightarrow \begin{array}{cc} \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} \end{array} \rightarrow \left[\begin{array}{c||c} \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} \end{array}\right] \rightarrow \begin{array}{cc} \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} \end{array}.$$

2. Turn over the two cards in the first row. If $\boxed{\heartsuit}\boxed{\clubsuit}$ appears, switch the left and right columns. Then, we obtain two commitments to x:

$$\begin{array}{rcc} & \boxed{\clubsuit} & \boxed{\heartsuit} \\ x: & \boxed{?} & \boxed{?} \\ x: & \boxed{?} & \boxed{?} \end{array}.$$

This is the copy protocol, which uses one pile-scramble shuffle. The revealed two $\boxed{\clubsuit}\boxed{\heartsuit}$ are now free cards.

2.4 AND-OR Protocol

The existing AND-OR Protocol [2] takes commitments to $a, b \in \{0,1\}$ as input and outputs two commitments to $a \wedge b$ and $a \vee b$ simultaneously using two free cards:

$$\underbrace{\boxed{?}\boxed{?}}_{a}\underbrace{\boxed{?}\boxed{?}}_{b}\boxed{\clubsuit}\boxed{\heartsuit} \Rightarrow \underbrace{\boxed{?}\boxed{?}}_{a \wedge b}\underbrace{\boxed{?}\boxed{?}}_{a \vee b}\boxed{\clubsuit}\boxed{\heartsuit}.$$

This protocol uses two pile-scramble shuffles; due to the page limitation, the details are omitted.

2.5 One-Input-Preserving AND Protocol

The existing one-input-preserving AND protocol [6] takes commitments to $a, b \in \{0,1\}$ as input and outputs a commitment to $a \wedge b$ while preserving the commitment to a using four free cards:

$$\underbrace{\boxed{?}\boxed{?}}_{a}\underbrace{\boxed{?}\boxed{?}}_{b}\boxed{\clubsuit}\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\heartsuit} \Rightarrow \underbrace{\boxed{?}\boxed{?}}_{a}\underbrace{\boxed{?}\boxed{?}}_{a \wedge b}\boxed{\clubsuit}\boxed{\heartsuit}\underbrace{\boxed{?}\boxed{?}}_{\overline{a} \wedge b}.$$

This protocol uses one pile-scramble shuffle. The commitment to $\overline{a} \wedge b$ is also obtained, but we do not use this commitment in our method and call the rightmost two cards *garbage* ones.

3 Our Method

In this section, we provide a method for determining whether a challenge in Tagiron is a success or a failure without leaking any information. As mentioned before, the proposed method consists of the *preprocessing protocol* and the *challenge protocol*. We give an overview in Section 3.1, and then Sections 3.2 and 3.3 give the preprocessing protocol and the challenge protocol, respectively.

3.1 Outline

In our proposed method, the preprocessing protocol (Section 3.2) will first create a commitment to $c_i \in \{0, 1\}$ above a numbered card $\boxed{i}$ for every $i \in \{0, 1, \ldots, 19\}$:

$$\begin{array}{ccccc} \left.\begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix}\right\} c_0 & \left.\begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix}\right\} c_1 & \left.\begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix}\right\} c_2 & \ldots & \left.\begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix}\right\} c_{19} \\ \boxed{0} & \boxed{1} & \boxed{2} & & \boxed{19} \end{array},$$

such that $c_i = 1$ if and only if $\boxed{i}$ is a target card. (Note that, of these 20 commitments, only the four have the value 1.)

Once such a sequence of commitments and numbered cards is obtained, any player who makes a challenge will execute the challenge protocol (Section 3.3). It can secretly extract the four commitments placed above the numbered cards corresponding to the player's guess and can perform a secure AND computation of those, i.e., obtain only the AND value of the four commitments without leaking any information (more than necessary). If the output of AND is 1, then all the guessed cards are target ones, and hence, the challenge is successful. If the output of AND is 0, the challenge fails.

There is one issue that should be resolved: remember that both the numbered cards $\boxed{10}$ and $\boxed{11}$ correspond to the Tagiron card $\boxed{\text{G5}}$. Therefore, if a player's guess contains exactly one $\boxed{\text{G5}}$, they cannot decide whether $\boxed{10}$ or $\boxed{11}$ is chosen. Thus, we want to be able to always choose $\boxed{10}$ in such a case. To this end, we swap the two commitments to c_{10} and c_{11} if $c_{10} < c_{11}$; we leave as they are if $c_{10} \geq c_{11}$. The concrete procedure is to create commitments to $d_{10} = c_{10} \vee c_{11}$ and $d_{11} = c_{10} \wedge c_{11}$ in the preprocessing protocol, and replace these new commitments with the commitments above $\boxed{10}$ and $\boxed{11}$:

$$\begin{array}{cccccc} \ldots & \left.\begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix}\right\} c_9 & \left.\begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix}\right\} d_{10} & \left.\begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix}\right\} d_{11} & \left.\begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix}\right\} c_{12} & \ldots. \\ & \boxed{9} & \boxed{10} & \boxed{11} & \boxed{12} & \end{array} \tag{3}$$

In the succeeding subsections, we present the preprocessing protocol and the challenge protocol.

3.2 Preprocessing Protocol

As seen in Section 1.3, at the beginning of a game, numbered cards are dealt as:

$$\begin{array}{l} P_1 : \boxed{?}\boxed{?}\boxed{?}\boxed{?},\ P_2 : \boxed{?}\boxed{?}\boxed{?}\boxed{?},\ P_3 : \boxed{?}\boxed{?}\boxed{?}\boxed{?},\ P_4 : \boxed{?}\boxed{?}\boxed{?}\boxed{?}, \\ \text{target cards} : \boxed{?}\boxed{?}\boxed{?}\boxed{?}. \end{array}$$

Given such 20 face-down numbered cards along with 21 $\boxed{\clubsuit}$ cards and 21 $\boxed{\heartsuit}$ cards (where 20 pairs of $\boxed{\clubsuit}$ and $\boxed{\heartsuit}$ will be used in Step 1 of the protocol and the remaining one pair will be used in Step 4), the following preprocessing protocol produces the sequence (3) in Section 3.1.

Preprocessing protocol

1. Place 20 $\boxed{\clubsuit}$ cards and 20 $\boxed{\heartsuit}$ cards as follows, so that we have a commitment to 0 above each of players' cards and a commitment to 1 above each target card:

$$\begin{array}{cccccccccccccccccccc}
 & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} & & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} & & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} & & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} \\
 & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} & & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} & & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} & & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} \\
P_1 : & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & P_2 : & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & P_3 : & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & P_4 : & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?}
\end{array}$$

$$\begin{array}{rcccc}
 & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} \\
 & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit}\,. \\
\text{target cards} : & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?}
\end{array}$$

2. After turning over all the face-up cards, regard each vertical 3-card sequence as a pile, and apply a pile-scramble shuffle to the 20 piles:

$$\left[\begin{array}{c|c|c|c|c|c|c} \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} \end{array}\right].$$

3. Reveal all the cards in the third row, i.e., the numbered cards, and arrange them in ascending order based on the revealed numbers while maintaining each pile of three cards:

$$\begin{array}{ccccccccccc} \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?}\,. \\ \boxed{0} & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} & \cdots & \boxed{10} & \boxed{11} & \cdots & \boxed{18} & \boxed{19} \end{array}$$

4. With the two commitments associated to $\boxed{10}$ and $\boxed{11}$ as inputs, apply the AND-OR protocol introduced in Section 2.4; then, the output of OR is newly associated to $\boxed{10}$ and the output of AND is newly associated to $\boxed{11}$.

The preprocessing protocol uses three shuffles in total.

3.3 Challenge Protocol

In this subsection, we present the challenge protocol, by which all players can know only the success or failure of a challenge. Suppose that the preprocessing protocol has been completed; then, given the sequence (3) along with 16 $\boxed{\clubsuit}$ cards and four $\boxed{\heartsuit}$ cards (where the player's guessed numbers will be identified by four $\boxed{\heartsuit}$ cards), the protocol proceeds as follows, where P_i is a player who guesses four cards and makes a challenge.

Challenge protocol

1. The player P_i holds 16 $\boxed{\clubsuit}$ cards and four $\boxed{\heartsuit}$ cards. Place the four $\boxed{\heartsuit}$ cards face down under the guessed numbered cards and $\boxed{\clubsuit}$ cards face down under the other numbered cards:

$$\begin{array}{ccccccccccc}
\boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} \\
\boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} \\
\boxed{0} & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} & \cdots & \boxed{10} & \boxed{11} & \cdots & \boxed{18} & \boxed{19} \\
\boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?}
\end{array}.$$

2. Turn over all the numbered cards. Apply a pile-scramble shuffle to the 20 piles:

$$\left[\begin{array}{c|c|c|c|c|c|c}
\boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} \\
\boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} \\
\boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} \\
\boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?}
\end{array}\right].$$

3. Turn over the 20 cards in the fourth row; then, four $\boxed{\heartsuit}$ cards appear at random positions:

$$\begin{array}{ccccccccccc}
\boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\
\boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\
\boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\
\boxed{\clubsuit} & \boxed{\heartsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\heartsuit} & \cdots & \boxed{\clubsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\clubsuit} & \boxed{\clubsuit}
\end{array}.$$

4. Let $c_{i_1}, c_{i_2}, c_{i_3}, c_{i_4}$ be the values of the four commitments associated to the four piles with $\boxed{\heartsuit}$; remember that we want to know only the value of $c_{i_1} \wedge c_{i_2} \wedge c_{i_3} \wedge c_{i_4}$. By the following procedure, we will obtain a commitment to $c_{i_1} \wedge c_{i_2} \wedge c_{i_3} \wedge c_{i_4}$ using eight free cards while keeping the original four commitments:

$$\underbrace{\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}}_{c_{i_1}} \; \underbrace{\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}}_{c_{i_2}} \; \underbrace{\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}}_{c_{i_3}} \; \underbrace{\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}}_{c_{i_4}} \; \begin{array}{cccc}\boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit} & \boxed{\clubsuit}\\ \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit} & \boxed{\heartsuit}\end{array}$$

$$\rightarrow \underbrace{\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}}_{c_{i_1}} \; \underbrace{\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}}_{c_{i_2}} \; \underbrace{\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}}_{c_{i_3}} \; \underbrace{\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}}_{c_{i_4}} \quad \underbrace{\begin{array}{c}\boxed{?}\\ \boxed{?}\end{array}}_{c_{i_1} \wedge c_{i_2} \wedge c_{i_3} \wedge c_{i_4}} . \qquad (4)$$

(a) Apply the copy protocol described in Section 2.3 to c_{i_1} to obtain two commitments to c_{i_1}. (Of the eight free cards, four are used, two are added, and the remaining number of free cards is six. The increase or decrease of free cards is described in the same way in the following steps.)
(b) With the commitments to c_{i_2} and c_{i_1} as input, apply the one-input-preserving AND protocol described in Section 2.5 to obtain commitments to $c_{i_1} \wedge c_{i_2}$ and c_{i_2}. (4 used, 2 added, 4 remaining, 2 garbage)
(c) With c_{i_3} and $c_{i_1} \wedge c_{i_2}$ as input, apply the one-input-preserving AND protocol to obtain $c_{i_1} \wedge c_{i_2} \wedge c_{i_3}$ and c_{i_3}. (4 used, 2 added, 2 remaining, 4 garbage)
(d) Since there are not enough free cards, shuffle the four garbage cards generated by the one-input-preserving AND protocol in steps (b) and (c) so far to make them free. (4 added, 6 remaining)
(e) With c_{i_4} and $c_{i_1} \wedge c_{i_2} \wedge c_{i_3}$ as input, apply the one-input-preserving AND protocol to obtain $c_{i_1} \wedge c_{i_2} \wedge c_{i_3} \wedge c_{i_4}$ and c_{i_4}. (4 used, 2 added, 4 remaining, 2 garbage)
(f) Shuffle the two garbage cards to set them free.

5. Step 4 yields the sequence (4) above. The commitments to $c_{i_1}, c_{i_2}, c_{i_3}, c_{i_4}$ are returned to their original positions. The commitment to $c_{i_1} \wedge c_{i_2} \wedge c_{i_3} \wedge c_{i_4}$ is turned over; the challenge succeeds if the value is 1 and fails if the value is 0.
6. Apply a pile-scramble shuffle to the 20 piles, turn over only the numbered cards, and arrange the piles in ascending order according to the revealed numbers. The sequence (3) is now reverted.

The number of shuffles required for one execution of the challenge protocol is eight. This consists of one pile-scramble shuffle in Step 2, the copy protocol once in Step 4, the one-input-preserving AND protocol three times, two garbage shuffles, and one pile-scramble shuffle in Step 5.

4 Generalization

Let U be the set of (numbers of) numbered cards, Y be the set of target numbers, and X be the set of guessed numbers, then the proposed method in Section 3 can be regarded as a secure computation protocol for determining whether $X = Y$ for secret sets $X, Y \subseteq U$ such that $|X| = |Y| = 4$, where $U = \{0, 1, \ldots, 19\}$ is public (and the set X can be chosen many times).

In the case of the 3-player version of Tagiron, it suffices to set $|X| = |Y| = 5$ since there will be five target cards, and it is easy to modify the protocols as such.

More generally, for any sets U, Y, X such that $Y, X \subseteq U$, it is easy to see that a secure computation protocol for determining whether $X \subseteq Y$ can be obtained by a modification of the proposed method.

5 Conclusion

In this paper, we proposed a novel method for applying card-based cryptography to the game of Tagiron. That is, we proposed a method for securely determining whether a set of expected elements is included in a target set without leaking any information more than necessary.

The proposed card-based protocols are simple and easy to understand for non-experts. Thus, it is useful as a tool to promote intuitive understanding of secure computations, and is expected to have educational value. More directly, this work could introduce the concept of secure computation to the Tagiron user community.

The number of cards used in our method is not small, and we admit that it is not practical. When we executed the protocols in a real-world setting (as shown in Figure 5), we observed a runtime of over one hour. Therefore, further optimization to reduce the number of cards is necessary and will be addressed in future work.

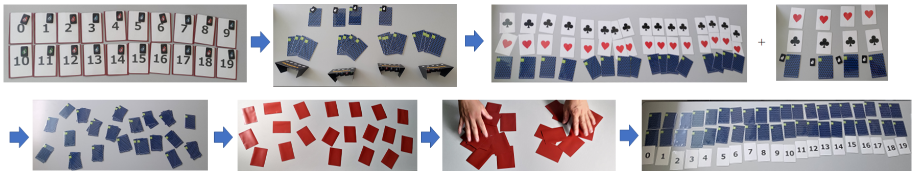

Fig. 5: Execution of the protocols in a real-world setting

Acknowledgements

We thank the anonymous reviewers, whose comments have helped us improve the presentation of the paper. This work was supported in part by JSPS KAKENHI Grant Numbers JP24K02938 and JP23H00479.

References

1. Boer, B.D.: More efficient match-making and satisfiability the five card trick. In: Quisquater, J.J., Vandewalle, J. (eds.) Advances in Cryptology – EUROCRYPT' 89. LNCS, vol. 434, pp. 208–217. Springer, Heidelberg (1990), https://doi.org/10.1007/3-540-46885-4_23
2. Francis, D., Aljunid, S.R., Nishida, T., Hayashi, Y., Mizuki, T., Sone, H.: Necessary and sufficient numbers of cards for securely computing two-bit output functions. In: Phan, R.C.W., Yung, M. (eds.) Paradigms in Cryptology—Mycrypt 2016. Malicious and Exploratory Cryptology. LNCS, vol. 10311, pp. 193–211. Springer, Cham (2017), https://doi.org/10.1007/978-3-319-61273-7_10
3. Fukasawa, T., Manabe, Y.: Card-based zero-knowledge proof for the nearest neighbor property: Zero-knowledge proof of ABC end view. In: Batina, L., Picek, S., Mondal, M. (eds.) Security, Privacy, and Applied Cryptography Engineering. LNCS, vol. 13783, pp. 147–161. Springer, Cham (2022), https://doi.org/10.1007/978-3-031-22829-2_9
4. Hand, S., Koch, A., Lafourcade, P., Miyahara, D., Robert, L.: Efficient card-based ZKP for single loop condition and its application to Moon-or-Sun. New Gener. Comput. **42**, 449–477 (2024), https://doi.org/10.1007/s00354-024-00274-1
5. Ikeda, S., Shinagawa, K.: How to play Mastermind without game master. In: Theory and Applications of Models of Computation. LNCS, Springer, Cham (2025, to appear)

6. Ishikawa, R., Chida, E., Mizuki, T.: Efficient card-based protocols for generating a hidden random permutation without fixed points. In: Calude, C.S., Dinneen, M.J. (eds.) Unconventional Computation and Natural Computation. LNCS, vol. 9252, pp. 215–226. Springer, Cham (2015), https://doi.org/10.1007/978-3-319-21819-9_16
7. Iwamoto, C., Ohara, K.: Card-based zero-knowledge proof for Dosun-Fuwari. IEICE Trans. Fundam. (2025), https://doi.org/10.1587/transfun.2024DML0001
8. Koch, A.: The landscape of security from physical assumptions. In: IEEE Information Theory Workshop. pp. 1–6. IEEE, NY (2021), https://doi.org/10.1109/ITW48936.2021.9611501
9. Komano, Y., Mizuki, T.: Physical zero-knowledge proof protocols for Topswops and Botdrops. New Gener. Comput. **42**, 399–428 (2024), https://doi.org/10.1007/s00354-024-00272-3
10. Komano, Y., Mizuki, T.: Card-based zero-knowledge proof protocols for pancake sorting. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences (2025, to appear). https://doi.org/10.1587/transfun.2025CIP0028
11. Miyahara, D., Robert, L., Lafourcade, P., Mizuki, T.: ZKP protocols for Usowan, Herugolf, and Five Cells. Tsinghua Science and Technology **29**(6), 1651–1666 (2024), https://doi.org/10.26599/TST.2023.9010153
12. Mizuki, T.: Preface: Special issue on card-based cryptography. New Gener. Comput. **39**, 1–2 (2021), https://doi.org/10.1007/s00354-021-00127-1
13. Mizuki, T.: Preface: Special issue on card-based cryptography 2. New Gener. Comput. **40**, 47–48 (2022), https://doi.org/10.1007/s00354-022-00170-6
14. Mizuki, T.: Preface: Special issue on card-based cryptography 3. New Gener. Comput. **42**, 303–304 (2024), https://doi.org/10.1007/s00354-024-00280-3
15. Mizuki, T., Kuzuma, T., Hirano, T., Oshima, R., Yasuda, M.: Gakmoro: An application of physical secure computation to card game. In: Unconventional Computation and Natural Computation. LNCS, Springer, Cham (2025, to appear)
16. Mizuki, T., Sone, H.: Six-card secure AND and four-card secure XOR. In: Deng, X., Hopcroft, J.E., Xue, J. (eds.) Frontiers in Algorithmics. LNCS, vol. 5598, pp. 358–369. Springer, Berlin, Heidelberg (2009), https://doi.org/10.1007/978-3-642-02270-8_36
17. Ono, T., Ruangwises, S., Abe, Y., Hatsugai, K., Iwamoto, M.: Single-shuffle physical zero-knowledge proof for sudoku using interactive inputs. In: Emura, K., Morita, H. (eds.) ACM ASIA Public-Key Cryptography Workshop. ACM, New York (2025)
18. Robert, L., Miyahara, D., Lafourcade, P., Mizuki, T.: Physical ZKP protocols for Nurimisaki and Kurodoko. Theor. Comput. Sci. **972**, 114071 (2023), https://doi.org/10.1016/j.tcs.2023.114071
19. Ruangwises, S.: NP-completeness and physical zero-knowledge proofs for Zeiger. In: Nakano, S.i., Xiao, M. (eds.) WALCOM: Algorithms and Computation. LNCS, vol. 15411, pp. 312–325. Springer, Cham (2025), https://doi.org/10.1007/978-981-96-2845-2_20
20. Ruangwises, S., Shinagawa, K.: Simulating virtual players for UNO without computers. In: Unconventional Computation and Natural Computation. LNCS, Springer, Cham (2025, to appear)
21. Shinagawa, K., Miyahara, D., Mizuki, T.: How to play Old Maid with virtual players. Theory of Computing Systems **69**(1) (2025), https://doi.org/10.1007/s00224-024-10203-w
22. Tamura, Y., Suzuki, A., Mizuki, T.: Card-based zero-knowledge proof protocols for the 15-puzzle and the token swapping problem. In: ACM ASIA Public-Key Cryptography Workshop. pp. 11–22. ACM, New York (2024), https://doi.org/10.1145/3659467.3659905
23. Tanaka, K., Sasaki, S., Shinagawa, K., Mizuki, T.: Only two shuffles perform card-based zero-knowledge proof for Sudoku of any size. In: 2025 Symposium on Simplicity in Algorithms (SOSA). pp. 94–107. SIAM (2025), https://doi.org/10.1137/1.9781611978315.7