Card-Based ZKP for Connectivity: Applications to Nurikabe, Hitori, and Heyawake*

Léo Robert[1*], Daiki Miyahara[2,3*], Pascal Lafourcade[1*]
and Takaaki Mizuki[4,3*]

[1]LIMOS, University Clermont Auvergne, Aubière, France.
[2]The University of Electro-Communications, Tokyo, Japan.
[3]National Institute of Advanced Industrial Science and
Technology (AIST), Tokyo, Japan.
[4]Cyberscience Center, Tohoku University, Sendai, Japan.

*Corresponding author(s). E-mail(s): leo.robert@uca.fr;
miyahara@uec.ac.jp; pascal.lafourcade@uca.fr;
mizuki@cc.tohoku.ac.jp;

Abstract

During the last years, several card-based Zero-Knowledge Proof (ZKP) protocols for Nikoli's puzzles have been designed. Although there are relatively simple card-based ZKP protocols for a number of puzzles, such as Sudoku and Kakuro, some puzzles face difficulties in designing simple protocols. For example, Slitherlink requires novel and elaborate techniques to construct a protocol. In this study, we focus on three Nikoli puzzles: Nurikabe, Hitori, and Heyawake. To date, no card-based ZKP protocol for these puzzles has been developed, partially because they have a relatively tricky rule that colored cells should form a connected area (namely, a polyomino); this rule, sometimes referred to as "Bundan-kin" (in Japanese), complicates the puzzles, as well as facilitating difficulties in designing card-based ZKP protocols.

We address this challenging task and propose a method for verifying the connectivity of hidden colored cells in a ZKP manner, such that we construct card-based ZKP protocols for the three puzzles.

Keywords: Zero-knowledge proofs, Card-based secure two-party protocols, Puzzle, Nurikabe, Hitori, Heyawake, Connectivity

1 Introduction

Zero-Knowledge Proofs (ZKP) were introduced by Goldwasser et al. [8]. This protocol involves two parties: a prover P and a verifier V. The prover P attempts to convince the verifier V that P possesses the solution s of a problem, without revealing any information about s. A ZKP protocol must satisfy the following three properties:

Completeness. If P knows the solution s, then P can convince V.
Soundness. If P does not know s, then P cannot convince V.
Zero-Knowledge. V learns nothing about s. Formally, simulated and actual protocol outputs follow the same probability distribution.

It was proven that for any NP-complete problem, there exists an interactive ZKP protocol [6]. Usually, ZKP protocols are executed on computers. In contrast, a *physical* ZKP protocol solely adopts physical algorithms with daily objects such as cards, envelopes, and bags, while prohibiting large computations (*i.e.*, no computer is allowed). In 2009, the first physical ZKP protocol was introduced for Sudoku [9], which is the most famous Nikoli* puzzle. Subsequently, efficient physical ZKP protocols for pencil puzzles have been proposed: Sudoku [25, 29], Nonogram [3, 24], Akari [1], Takuzu [1, 17], Kakuro [1, 16], Kenken [1], Makaro [2], Norinori [5], Slitherlink [15], Juosan [17], Suguru [23], Ripple Effect [27], Numberlink [26], Bridges [28], and Cryptarithmetic [13]. All these physical protocols employ a deck of cards as physical objects. Hence, we call such a protocol a *card-based* ZKP protocol hereinafter.

In this study, we address three other Nikoli puzzles: *Nurikabe*, *Hitori*, and *Heyawake*[†], with a common rule. Examples of their cases and solutions are illustrated in Figures 1, 2, and 3. The common rule is about a geometric structure that requires white cells (in a solution) to form a connected area, namely, a *polyomino*. This rule, sometimes referred to as "Bundan-kin" (in Japanese), complicates the puzzles, and causes difficulties in designing simple card-based ZKP protocols.

---

*Nikoli is a game publisher famously known for its Sudoku puzzle as well as several other pencil puzzles.

[†]Solving these three puzzles is proved to be NP-complete. In [12], solving even simple versions of Nurikabe was proven to be NP-complete. In [10], the authors proved that Hitori is NP-complete. In [11], solving a Heyawake puzzle is proved to be NP-complete.

Connectivity constraint

Let us discuss this rule in a ZKP approach. Assume that a prover P has a polyomino in mind comprising white cells (as illustrated in the solutions in Figures 1, 2, and 3), which we call a *white-polyomino*, on a grid, while a verifier solely knows the grid. The prover P attempts to convince the verifier V that P knows the white-polyomino without revealing it. Accordingly, a novel technique is required, as demonstrated later. Hereinafter, we refer to this common rule as the *connectivity* constraint.

Contributions

In this study, we construct card-based ZKP protocols for Nurikabe, Hitori, and Heyawake. To achieve this, we present a generic method to address the connectivity constraint. Specifically, after introducing some of the known techniques in Section 2, we present the generic method to verify the connectivity constraint, together with a simple observation. Then, using the generic method, we construct card-based ZKP protocols for Nurikabe, Hitori, and Heyawake in Sections 4, 5, and 6, respectively. Finally, the paper is concluded in Section 7.

One might think that physical ZKP protocols for these three puzzles could be constructed by transforming a known physical ZKP protocol for an NP-complete problem, such as a lockable-box-based ZKP protocol for 3-Colorability [7]; however, such a transformation is infeasible because 1) the overhead must be included in the transformation, 2) it is challenging to construct a method for transforming an instance of a puzzle into a graph whose 3-colorability respects the solvability of the puzzle[‡], and 3) the transformed ZKP protocol does not capture the property of a puzzle. Therefore, we want to develop a direct (and specialized) card-based ZKP protocol for each of them.

This study is inspired by [15], which focused on Slitherlink and Masyu, where P has to convince V of a single loop property. We follow a similar strategy to [15]. That is, P first colors cells successively, such that the resulting cells are guaranteed to satisfy the connectivity constraint; then, V verifies all the remaining constraints. We note that the proposed protocols could not be constructed by simply adopting the existing technique [15].

An earlier version of this paper was presented and appeared as a conference paper [22]. The primary difference between the two is as follows. First, we have provided a generic method to address pencil puzzles with the connectivity constraint, including the idea behind our method in Section 3. Second, we have added Section 6 to present a card-based ZKP protocol for Heyawake using the generic method (whereas the conference paper solely handled Nurikabe and Hitori). Third, Sections 4.3 and 5.3 have been enhanced to provide full security proofs (whereas the conference paper just provided sketches).

[‡]Conversely, transforming a graph into a pencil puzzle has already been proposed as in the NP-completeness proof [12].
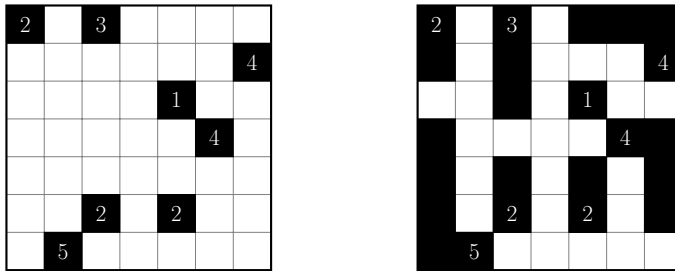
Fig. 1 Initial Nurikabe grid on the left and its solution on the right.
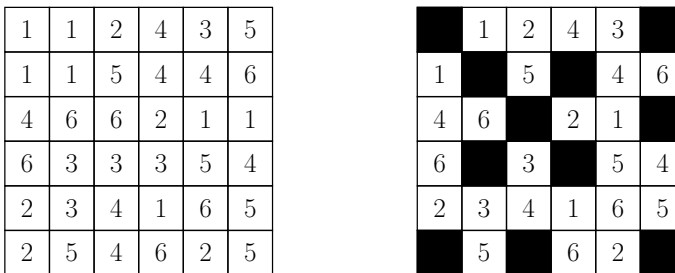
Fig. 2 Initial Hitori grid on the left and its solution on the right.

Nurikabe's rule

This puzzle[§] is played on a rectangular grid, where some cells contain numbers (Figure 1). The objective is to color some cells[¶] in black, such that the following conditions are met:

1. Each numbered cell identifies the number of continuous black cells (separated by white cells). This region comprising black cells is called an *island*.
2. An island must contain only one numbered cell.
3. The white cells form a white-polyomino (called a *sea*).
4. The sea cannot form a 2 × 2 area.

Hitori's rule

This puzzle[‖] comprises a grid and numbers where each cell contains a single number, as demonstrated in Figure 2. The objective is to color some cells in black, with the following constraints:

1. Each row and each column must contain only one occurrence of a number (on white cells).
2. Two black cells cannot share a side; however, they can share a corner vertex.
3. The numbered cells (white cells) must form a white-polyomino.

[§]https://www.nikoli.co.jp/en/puzzles/nurikabe.html.
[¶]The original puzzle has colors white and black exchanged; we choose to reverse them for coherence with regard to the other puzzles, *i.e.*, Hitori and Heyawake.
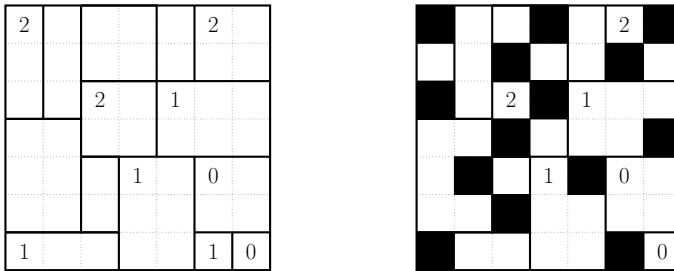[‖]https://www.nikoli.co.jp/en/puzzles/hitori.html.

Fig. 3 Initial Heyawake grid on the left and its solution on the right.

Heyawake's rule

This puzzle[**] (as illustrated in Figure 3) is a grid comprising white cells and numbered cells. The grid is delimited by regions called *rooms*. The objective is to color some cells in black with the following constraints:

1. Each room with a number has the corresponding number of black cells in it. A room with no number can have any number of black cells.
2. Two black cells cannot share a side; however, they can share a corner vertex.
3. The white cells must form a white-polyomino.
4. White cells cannot stretch across more than two rooms in a straight line.

2 Preliminaries

In this section, we introduce some card and shuffle notations, and explain some of the existing card-based sub-protocols adopted in our constructions later.

Card

In our protocols, a deck of cards comprises clubs ♣ ♣ ⋯ , hearts ♡ ♡ ⋯ , and number cards 1 2 ⋯ , whose backs are identical ? . We encode three colors {black, white, red} with the order of two cards as follows:

$$\boxed{♣}\,\boxed{♡} \to \text{ black}, \quad \boxed{♡}\,\boxed{♣} \to \text{ white}, \quad \boxed{♡}\,\boxed{♡} \to \text{ red}. \tag{1}$$

We call a pair of face-down cards ? ? corresponding to a color according to the above encoding rule a *commitment* to the respective color. We also use the terms, *black commitment*, *white commitment*, and *red commitment*. We sometimes regard black and white commitments as bit values, based on the following encoding:

$$\boxed{♣}\,\boxed{♡} \to 0, \quad \boxed{♡}\,\boxed{♣} \to 1. \tag{2}$$

[**] https://www.nikoli.co.jp/en/puzzles/heyawake.html.

For a bit $x \in \{0, 1\}$, if a pair of face-down cards satisfies the encoding (2), we say that it is a commitment to x, denoted by

$$\underbrace{\boxed{?}\,\boxed{?}}_{x}.$$

Pile-shifting shuffle [20, 30]

This shuffling action means to *cyclically* shuffle piles of cards. More formally, given m piles, each of which consists of the same number of face-down cards, denoted by $(\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_m)$, applying a *pile-shifting shuffle* (denoted by $\langle \cdot \| \cdots \| \cdot \rangle$), results in $(\mathbf{p}_{s+1}, \mathbf{p}_{s+2}, \ldots, \mathbf{p}_{s+m})$:

$$\left\langle \underbrace{\boxed{?}}_{\mathbf{p}_1} \middle\| \underbrace{\boxed{?}}_{\mathbf{p}_2} \middle\| \cdots \middle\| \underbrace{\boxed{?}}_{\mathbf{p}_m} \right\rangle \rightarrow \underbrace{\boxed{?}}_{\mathbf{p}_{s+1}} \underbrace{\boxed{?}}_{\mathbf{p}_{s+2}} \cdots \underbrace{\boxed{?}}_{\mathbf{p}_{s+m}},$$

where s is uniformly and randomly selected from $\mathbb{Z}/m\mathbb{Z}$. Implementing a pile-shifting shuffle is simple: we adopt physical cases that can store a pile of cards, such as boxes and envelopes. Here, a player (or players) cyclically shuffles them manually until everyone (*i.e.*, P and V) loses track of the offset. Note that "everyone" is an essential assumption for security.

Chosen pile protocol [5]

This protocol is an extended version of the "chosen pile cut" proposed in [14]. This protocol allows a prover P and a verifier V to obtain a pile of cards P chose from multiple piles, without V knowing which one it is. Some operations can be manipulated on this pile, and all the piles, including this pile, can be replaced in their initial order.

Precisely, given m piles $(\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_m)$ with $2m$ additional cards, the *chosen pile protocol* enables P and V to obtain the i-th pile $\mathbf{p}_i$ P chose (without revealing the index i) and reverts the sequence of m piles to their original order after applying other operations to $\mathbf{p}_i$.

1. Using $m - 1$ $\boxed{\clubsuit}$s and one $\boxed{\heartsuit}$, P places m face-down cards (denoted by *row 2*) below the given piles, such that only the i-th card is $\boxed{\heartsuit}$. We further position m cards (denoted by *row 3*) below the cards, such that only the first card is $\boxed{\heartsuit}$:

$$\underbrace{\boxed{?}}_{\mathbf{p}_1} \underbrace{\boxed{?}}_{\mathbf{p}_2} \cdots \underbrace{\boxed{?}}_{\mathbf{p}_{i-1}} \underbrace{\boxed{?}}_{\mathbf{p}_i} \underbrace{\boxed{?}}_{\mathbf{p}_{i+1}} \cdots \underbrace{\boxed{?}}_{\mathbf{p}_m},$$

$$\begin{array}{ccccccc} \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} \\ \clubsuit & \clubsuit & & \clubsuit & \heartsuit & \clubsuit & & \clubsuit \end{array} \leftarrow \text{row 2,}$$

$$\begin{array}{ccccccc} \boxed{?} & \boxed{?} & \cdots & \boxed{?} & \boxed{?} & \boxed{?} & \cdots & \boxed{?} \\ \heartsuit & \clubsuit & & \clubsuit & \clubsuit & \clubsuit & & \clubsuit \end{array} \leftarrow \text{row 3.}$$

2. Considering the cards in the same column as a pile, apply a pile-shifting shuffle to the sequence of piles.

3. Reveal all the cards in *row 2*. Consequently, exactly one $\heartsuit$ appears, and the pile above the revealed $\heartsuit$ is the i-th pile (hence, P can obtain $\mathbf{p}_i$). After this step is invoked, other operations are applied to the chosen pile. Then, the chosen pile is reverted back to the i-th position in the sequence.

4. Remove the revealed cards, *i.e.*, the cards in *row 2*. (Note that we do not use the card $\heartsuit$ revealed in Step 3.) Subsequently, apply a pile-shifting shuffle.

5. Reveal all the cards in *row 3*. Consequently, one $\heartsuit$ appears, and the pile above the revealed $\heartsuit$ is $\mathbf{p}_1$. Therefore, by shifting the sequence of piles (such that $\mathbf{p}_1$ becomes the leftmost pile in the sequence), we can obtain a sequence of piles whose order is the same as the original one without revealing any information about the order of the input sequence.

Input-preserving five-card trick [17]

Given two commitments to $a, b \in \{0,1\}$ based on the encoding rule (2), this sub-protocol [4, 17] reveals only the value of $a \vee b$ and restores commitments to a and b:

$$\underbrace{\boxed{?}\,\boxed{?}}_{a}\ \underbrace{\boxed{?}\,\boxed{?}}_{b} \ \rightarrow\ a \vee b\ \&\ \underbrace{\boxed{?}\,\boxed{?}}_{a}\ \underbrace{\boxed{?}\,\boxed{?}}_{b};$$

the original sub-protocol [4, 17] was designed to compute AND $(a \wedge b)$; however, we adjust it to compute OR $(a \vee b)$ in Sections 5 and 6.

The sub-protocol proceeds as follows.

1. Add helping cards and swap the two cards of the commitment to b to obtain the negation $\bar{b}$, as follows:

$$\underbrace{\boxed{?}\,\boxed{?}}_{a}\ \underbrace{\boxed{?}\,\boxed{?}}_{b}\ \rightarrow\ \underbrace{\boxed{?}\,\boxed{?}}_{a}\,\boxed{\heartsuit}\ \underbrace{\boxed{?}\,\boxed{?}}_{\bar{b}}\,\boxed{\heartsuit}\,\boxed{\clubsuit}\,\boxed{\clubsuit}\,\boxed{\clubsuit}\,\boxed{\clubsuit}.$$

2. Rearrange the sequence of cards and turn over the face-up cards as:

$$\boxed{?}\,\boxed{?}\,\boxed{\heartsuit}\,\boxed{?}\,\boxed{?}\,\boxed{\heartsuit}\,\boxed{\clubsuit}\,\boxed{\clubsuit}\,\boxed{\clubsuit}\,\boxed{\clubsuit}\ \rightarrow\ \begin{matrix}\boxed{?}\,\boxed{?}\,\boxed{\heartsuit}\,\boxed{?}\,\boxed{?}\\ \boxed{\heartsuit}\,\boxed{\clubsuit}\,\boxed{\clubsuit}\,\boxed{\clubsuit}\,\boxed{\clubsuit}\end{matrix}\ \rightarrow\ \begin{matrix}\boxed{?}\,\boxed{?}\,\boxed{?}\,\boxed{?}\,\boxed{?}\\ \boxed{?}\,\boxed{?}\,\boxed{?}\,\boxed{?}\,\boxed{?}\end{matrix}.$$

3. Regarding cards in the same column as a pile, apply a pile-shifting shuffle to the sequence:

$$\left\langle \begin{matrix}\boxed{?}\\\boxed{?}\end{matrix}\,\Big\|\,\begin{matrix}\boxed{?}\\\boxed{?}\end{matrix}\,\Big\|\,\begin{matrix}\boxed{?}\\\boxed{?}\end{matrix}\,\Big\|\,\begin{matrix}\boxed{?}\\\boxed{?}\end{matrix}\,\Big\|\,\begin{matrix}\boxed{?}\\\boxed{?}\end{matrix}\right\rangle \rightarrow \begin{matrix}\boxed{?}\,\boxed{?}\,\boxed{?}\,\boxed{?}\,\boxed{?}\\ \boxed{?}\,\boxed{?}\,\boxed{?}\,\boxed{?}\,\boxed{?}\end{matrix}.$$

4. Reveal all the cards in the first row.
   (a) If the resulting sequence is $\boxed{\clubsuit}\,\boxed{\clubsuit}\,\boxed{\heartsuit}\,\boxed{\heartsuit}\,\boxed{\heartsuit}$ (up to cyclic shifts), then $a \vee b = 0$.
   (b) If it is $\boxed{\heartsuit}\,\boxed{\clubsuit}\,\boxed{\heartsuit}\,\boxed{\clubsuit}\,\boxed{\heartsuit}$ (up to cyclic shifts), then $a \wedge b = 1$.

5. After turning over all the face-up cards, apply a pile-shifting shuffle.

6. Reveal the two cards in the second row; then, the revealed cards should include exactly one $\boxed{\heartsuit}$.
7. Shift the sequence of piles, such that the revealed $\boxed{\heartsuit}$ is the leftmost card, and swap the two cards of the commitment to $\bar{b}$, to restore commitments to a and b.

3 How to Form a White Polyomino

In this section, we present a generic method to support constructions of a card-based ZKP protocol for a pencil puzzle with the connectivity constraint, *i.e.*, white cells must form a white-polyomino. Using our method, we construct card-based ZKP protocols for our targets, *i.e.*, Nurikabe, Hitori, and Heyawake, in Sections 4, 5, and 6, respectively.

3.1 Idea

The idea behind our novel technique emerges from a simple observation. Suppose that we are given a rectangle grid where only one cell is white and the remaining cells are black. We will change the color of a black cell to white successively, as follows. If we color one of the four neighbor cells white around the initial white cell, then the resulting white cells should be connected and not separated. Therefore, if we repeatedly color one of the four neighbors white around any white cell, then we obtain a white-polyomino because we color only an adjacent cell around a white cell. We let the prover P perform this transformation of colors and make a desired polyomino to convince the verifier V that the resulting cells satisfy the connectivity constraint.

3.2 Subprotocol: 4-Neighbor Protocol

Before comprehensively describing our method, we present a subprotocol, called the *4-neighbor protocol* that is benefical to our generic construction. This protocol enables a prover P to indicate a specific commitment from commitments on a grid, including one of the four commitments next to the specific commitment without a verifier V knowing their exact positions.

Precisely, given pq commitments placed on a $p \times q$ grid, a prover P has an intended commitment, which we call a *target* commitment. The prover P attempts to reveal the target commitment and another one that lies next to the target commitment (without revealing their exact positions). Here, a verifier V should be convinced that the second commitment is a neighbor of the first one (without knowing which one), and V should be able to confirm the colors of both commitments. To handle the case where the target commitment is at the edge of the grid, we place commitments to red (as "dummy" commitments) on the left of the first column and below the last row, to prevent P from choosing a commitment that is not a neighbor. Accordingly, the size of the expanded grid is $(p+1) \times (q+1)$ (not $(p+2) \times (q+2)$, as will be explained in Step 3).

This subprotocol proceeds as follows.

1. P and V pick the $(p+1)(q+1)$ commitments on the grid from left-to-right and top-to-bottom to make a sequence of commitments:

$$\boxed{?}\boxed{?}\ \ \boxed{?}\boxed{?}\ \ \boxed{?}\boxed{?}\ \ \boxed{?}\boxed{?}\ \cdots\ \boxed{?}\boxed{?}.$$

2. P uses the chosen pile protocol (Section 2) to reveal the target commitment.
3. P and V pick all the four neighbors of the target commitment. Because a pile-shifting shuffle is a cyclic reordering, the distance between commitments are maintained (up to a given modulo). In other words, for a target commitment (not at the edge), the possible four neighbors are at a distance for the left or right one, and $p + 1$ for the bottom or top one. Therefore, P and V can determine the positions of all the four neighbors.

    Recall that dummy commitments are placed in the leftmost column and bottom row. If a target commitment is at the edge, for instance, top rightmost, its right commitment is the dummy at the second from the top leftmost, while its top commitment is the dummy at the bottom rightmost position. Consequently, a $(p + 1) \times (q + 1)$ grid suffices.
4. Among these four neighbors, P chooses one commitment using the chosen pile protocol and reveals it.
5. P and V end the second and first chosen pile protocols.

3.3 Method

Using the 4-neighbor protocol presented in Section 3.2, we present a generic method to perform the transformation of colors mentioned in Section 3.1, without revealing any information on the resulting cells.

Assume that there is a grid with $p \times q$ cells. P attempts to arrange white commitments on the grid, such that they form a white-polyomino while V is convinced that the placement of commitments is certainly a white-polyomino. Our generic method is as follows.

1. P and V place a commitment to black (*i.e.*, $\boxed{\clubsuit}\boxed{\heartsuit}$) on every cell and commitments to red as mentioned in Section 3.2, such that they have $(p + 1)(q + 1)$ commitments on the board.
2. P uses the chosen pile protocol to choose one black commitment that P attempts to change.
    (a) V swaps the two cards constituting the chosen commitment, such that it becomes a white commitment (recall the encoding (1)).
    (b) P and V end the chosen pile protocol to return the commitments to their original positions.
3. P and V repeat the following steps exactly $pq - 1$ times.
    (a) P chooses one white commitment as a target and one black commitment among its neighbors using the 4-neighbor protocol; the neighbor is chosen such that P attempts to make it white.
    (b) V reveals the target commitment. If it corresponds to white, then V continues; otherwise V aborts.

(c) V reveals the neighbor commitment (chosen by P). If it corresponds to black, then P makes the neighbor white or keeps it black (depending on P's choice) by executing the following steps; otherwise V aborts.

   (i) If P attempts to change the commitment, P places face-down club-to-heart pair below it; otherwise, P places a heart-to-club pair:

$$\boxed{?}\,\boxed{?} \;\rightarrow\; \begin{matrix}\boxed{?}\,\boxed{?} \\ \boxed{?}\,\boxed{?} \\ \clubsuit \;\; \heartsuit\end{matrix} \quad \text{or} \quad \begin{matrix}\boxed{?}\,\boxed{?} \\ \boxed{?}\,\boxed{?} \\ \heartsuit \;\; \clubsuit\end{matrix}.$$

  (ii) Regarding cards in the same column as a pile, V applies a pile-shifting shuffle to the sequence of piles:

$$\left\langle \begin{matrix}\boxed{?}\\\boxed{?}\end{matrix} \middle\| \begin{matrix}\boxed{?}\\\boxed{?}\end{matrix} \right\rangle \;\rightarrow\; \begin{matrix}\boxed{?}\,\boxed{?}\\\boxed{?}\,\boxed{?}\end{matrix}.$$

  (iii) V reveals the two cards in the second row. If the revealed right card is $\boxed{\heartsuit}$, then V swaps the two cards in the first row; otherwise V does nothing.

(d) P and V end the 4-neighbor protocol.

After this process, V is convinced that all the white commitments represent a white-polyomino. Therefore, this method enables a prover P to make a solution that only P has, guaranteed to satisfy the connectivity constraint.

If the number of white cells in the final polyomino, *e.g.*, k, is public to a verifier V, it is sufficient that in Step 3, P and V repeat $k - 1$ times and in Step 3c, V simply swaps the two cards comprising the neighbor commitment to make it white (without P's choice).

4 ZKP Protocol for Nurikabe

In this section, we propose a card-based ZKP protocol for Nurikabe. Our protocol comprises two phases: the setup and verification phases, which are presented in Sections 4.1 and 4.2, respectively. Its security proof is provided in Section 4.3.

Consider a puzzle instance of a $p \times q$ grid containing m numbered cells, such that the i-th numbered cell (in any order) has a number x_i for every i, $1 \le i \le m$. Recall that an island of a Nurikabe puzzle must contain exactly one numbered cell, and the number of black cells inside the island is indicated by the number written on the numbered cell. Therefore, the number of white cells in the solution (namely, in the white-polyomino), denoted by N_w, is the difference between the number of total and black cells (including the numbered cells). Therefore, we have

$$N_w = pq - \sum_{i=1}^{m} x_i.$$

This number N_w can be regarded as public information, and indeed, we explicitly use the number N_w in our protocol.

4.1 Setup Phase

The prover P (having a polyomino in mind as a solution) and the verifier V adopt the generic method presented in Section 3, where N_w is used as public information, such that Step 3 is repeated $N_w - 1$ times. The resulting configuration is a placement of $(p+1)(q+1)$ commitments on the board where the commitments in the leftmost column and the bottom row correspond to red, and the remaining commitments represent the solution according to the encoding (1) (because P makes the white-polyomino corresponding to the solution). Subsequently, V is convinced that all the white commitments form a white-polyomino, *i.e.*, the connectivity constraint is satisfied.

4.2 Verification Phase

V first verifies that the current commitments placed on the grid (after the setup phase) satisfy the rule 4 (forbidden 2×2 area). Then, V verifies the rules 1 and 2, relative to the black commitments (island constraints).

Sea rule: Forbidden area (rule 4)

The prover P attempts to convince V that any 2×2 area contains at least one black cell. Note that all 2×2 areas are determined, given an initial grid. Indeed, for a given $p \times q$ grid, there are $(p-1)(q-1)$ possible areas.

Hence, P and V consider each 2×2 area (consisting of four commitments) successively (in any order), and will repeat the following for each possible area.

1. P chooses a black commitment on this 2×2 area via the chosen-pile protocol applied to the four commitments.
2. V reveals the commitment chosen by P. If the revealed commitment corresponds to black, then V is convinced that the 2×2 area is not solely formed by white commitments. Otherwise, V aborts:

$$\boxed{?}\boxed{?}\;\boxed{?}\boxed{?} \;\to\; \text{Chosen pile protocol} \;\to\; \boxed{?}\boxed{?}\;\boxed{\clubsuit}\boxed{\heartsuit} \\ \boxed{?}\boxed{?}\;\boxed{?}\boxed{?} \qquad\qquad\qquad\qquad\qquad \boxed{?}\boxed{?}\;\boxed{?}\boxed{?}.$$

Island rules (rule 1, 2)

P attempts to convince V that the black cells respect the constraints. There are two verifications to consider: solely one numbered cell for a given region (island) and all black commitments are connected inside the island. These two constraints are verified in the following protocol.

Let $n \geq 2$ be the number written on a given numbered cell.[††] We are going to change the n continuous black commitments (on the island) into


[††]For a numbered cell where one is written, V simply reveals the commitment on it and its four neighbors to confirm that the island is surrounded by the sea.

red ones successively using the 4-neighbor protocol. Notice that we use red commitments to ensure that each black cell is checked exactly once.

1. V reveals the commitment on the numbered cell. If it corresponds to black, V replaces it with a red commitment; otherwise V aborts.
2. Repeat the following steps exactly $n - 1$ times.
   (a) P uses the 4-neighbor protocol to choose a red commitment as a target, and one black commitment among its neighbors.
   (b) V reveals the target commitment. If it corresponds to red, then V continues; otherwise V aborts.
   (c) V reveals the neighbor commitment (chosen by P). If it corresponds to black, V replaces it with a red commitment; otherwise V aborts.
   (d) P and V end the 4-neighbor protocol to return the commitments to their original positions.

Now, V is convinced that the size of the island comprising black cells is greater than or equal to n because Step 2 just indicates that there are $n-1$ continuous black cells adjacent to the number cell. To verify that the size of the island is equal to n, it suffices to prove that there exists no black cell around the island. Accordingly, we change all the n red commitments into white ones successively, and confirm that there is no black commitment around them. Notice that we can check each red commitment exactly once.

3. V replaces the commitment on the numbered cell with a white commitment.
4. Repeat the following steps exactly $n - 1$ times.
   (a) P uses the chosen pile protocol to choose a red commitment.
   (b) V reveals the chosen commitment. If it corresponds to red, V continues; otherwise V aborts.
   (c) Recall that P attempts to verify that any of four neighbor commitments is not black. Recall also the encoding (1), *i.e.*, the left card of a white or red commitment is a heart ♡. V reveals the left card of each of the four neighbors. If all of them are hearts (which means that all the commitments do not correspond to black), V replaces the chosen commitment with a white commitment; otherwise V aborts:
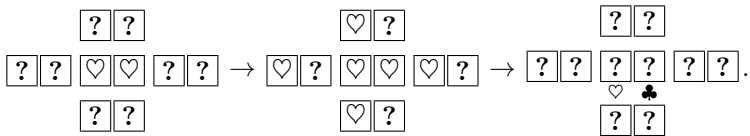
   (d) P and V end the chosen pile protocol to revert the commitments to their original positions.

By applying the aforementioned steps to all the numbered cells, V is convinced that the placement of the commitments satisfies all the constraints, *i.e.*, P possesses the solution.

4.3 Security Proofs

We present the following theorems to demonstrate that our protocol respects the security properties.

Theorem 1 (Completeness) *If P knows a solution of a Nurikabe grid, then P can convince V.*

Proof Suppose that P, which knows the solution s of the grid, runs the setup phase as in Section 4.1. We will demonstrate that P can perform the proofs for the sea formation and the verification phases.

Setup: In this phase, the goal of P is to prove that the white commitments form a white-polyomino (rule 3). Because s is a solution to the initial grid, the white cells appearing (in the solution) are connected. Hence, P can always pick a white commitment. The corresponding neighbor can be changed into a white commitment, thereby ensuring that the resulting figure is connected.

Verification: The verifier V must be convinced that rules 1, 2, and 4 hold. First, the *forbidden area* (rule 4) must be verified. Because s is a solution, there is no 2×2 area solely comprising white commitments on the grid. Therefore, by checking each 2×2 area, there will be at least one black commitment among the chosen four commitments on the area. Hence, P can choose (via the chosen-pile protocol) such a black commitment.

   Second, the *island rules* (rules 1 and 2) must be verified. Because s is a solution, each numbered cell is in a region (island) with the corresponding number of black commitments and no additional numbered cell. Consider a numbered cell equal to $n > 2$ (the case $n = 1$ is straightforward because it must have its four neighbors all in white, which can be verified by simply revealing them). There are $n - 1$ black commitments without counting the commitment on the numbered cell. Hence, P and V can apply the first sub-routine exactly $n - 1$ times. This implies that all the black commitments are turned into red ones. From here, V is convinced that the region contains at least n black commitments. However, this could be more than n, if two regions are not separated. Hence, the second sub-routine ensures that the region does not comprise more than n black commitments. Note that at this point, all black cells are encoded as red commitments: ♡ ♡ .

   Now, V replaces the commitment on the numbered cell by a white commitment. When revealing the left card of the four neighbors for a given previous black commitment, there are all ♡ because all previous black commitments are encoded as ♡ ♡ . The verified commitments are then turned into white ones; hence, V is convinced that no red commitment is checked twice.

   Finally, because all the verifications have been checked, we proved that if P has a solution, then the verifications will always succeed.   □

Theorem 2 (Soundness) *If P does not provide a solution of a Nurikabe grid, then P cannot convince V.*

Proof Suppose that P does not provide a solution for the grid in the setup phase. We will demonstrate that V will always detect it.

Suppose that the commitments are correct, meaning that the white commitments form a white-polyomino. (If it is not the case, then V aborts because a black commitment can be whitened if and only if one of its four neighbors is already whitened.)

We can distinguish two cases, each corresponding to a part of the verification phase.

- *Forbidden area.* V verifies all possible 2×2 areas of the grid. If one of them solely comprises only white commitments, then P cannot choose a black one. Therefore, V will reveal a white one, as P cannot act otherwise.

- *Island rules.* Consider a numbered cell equal to $n\,(> 2)$. We divide all the invalid cases into three, depending on the size of their islands. Each example is presented as follows:

(1)   (2)   (3)
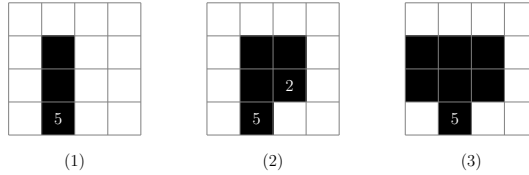
  1. Consider the size is less than n, *i.e.*, only $\ell\,(< n)$ black commitments constitute the island. During Step 2c for the ℓ-th instance, the neighbor commitment cannot correspond to a black commitment; thus, V aborts.
  2. Consider the size is equal to n, but some other numbered cells are inside the island. Note that when the verification is completed for the numbered cell equal to n, the commitments on the other numbered cells will become white ones. Consequently, in Step 1 a white commitment should appear when V verifies the other numbered cells, and then V aborts.
  3. Consider the size is greater than $n^{\ddagger\ddagger}$. This means that after Step 2 there are more black commitments around red commitments. V can detect it in Step 4c because a club $\boxed{\clubsuit}$ should appear when V reveals the left card of each of four neighbors of a red commitment.

Therefore, if P does not provide a solution, V always detects it.   □

Theorem 3 (Zero-knowledge) *V learns nothing about P's solution of a given Nurikabe grid.*

Proof We adopt the same proof technique as in [9]: zero-knowledge is induced by a description of an efficient *simulator* that simulates the interaction between a cheating verifier and a real prover. Here, although the simulator does not have a solution, it can swap cards for different ones during shuffles. The simulator acts as follows.

- The simulator constructs a random white-polyomino of size N_w.

---

‡‡We do not consider other numbered cells here because they did not affect whether V aborts or not.

- During the verification of the forbidden area rule, each possible 2×2 area is verified. For a given area, the corresponding commitments are shuffled; thus, the simulator can swap the target commitment by a black commitment via the chosen-pile protocol.

- During the verification of the island rule, the simulator replaces the target commitment by a red one, and its corresponding neighbor by a black one, for the first sub-routine.

  For the second sub-routine, the simulator changes the target commitment by a red commitment, and also change the left card of each of the four neighbors by heart cards.

Note that the chosen pile and 4-neighbor protocols use shuffle techniques; thus, the simulator can replace shuffled cards when applying these protocols. The simulated and real proofs are indistinguishable; hence, V learns nothing about P's solution.

□

5 ZKP Protocol for Hitori

In this section, we present a card-based ZKP protocol for Hitori. Our protocol comprises the setup and verification phases, which are presented in Sections 5.1 and 5.2, respectively. Its security proof is provided in Section 5.3.

5.1 Setup Phase

This phase follows the same steps as in the generic method presented in Section 3.3.

After the above process, V is convinced that the resulting commitments represent a white-polyomino (rule 3), and information on the number of white commitments is hidden from V.

5.2 Verification phase

One occurrence for each row/column (rule 1)

Here, V verifies if each row and column contains only one occurrence of a number (on the white commitments). The idea is that for a given row or column, it suffices to solely consider all cells with the same number that appear $k > 1$ times and confirm that the k commitments on the numbered cells correspond to either k blacks or $k - 1$ blacks without revealing its correspondence. For example, consider the first row in the puzzle instance of Figure 2, which is $(1, 1, 2, 4, 3, 5)$; because '1' appears twice, we check that the two commitments on '1' are both black or one of them is black (if both of them are white, then the rule is not satisfied).

For a given row or column, this verification proceeds as follows.

1. V looks for the same numbered cells that appear more than once; let k be the number of them. V picks the k commitments on them.
2. P uses the chosen pile protocol to choose a white commitment among the k commitments if it exists; otherwise, P chooses any commitment among them.

3. V reveals the $k-1$ commitments that are not chosen by P. If all of them correspond to black (this means that the k commitments correspond to k or $k-1$ blacks), V continues; otherwise V aborts.
4. P and V end the chosen pile protocol to return the k commitments to their original places.
5. P and V repeat the above steps for all numbered cells that appear more than once.

Lonely black (rule 2)

V verifies that black cells are isolated from each other. Recalling the encoding (2), we will compute logical ORs: A white commitment corresponds to bit 1 and a black commitment corresponds to bit 0. For each pair of adjacent commitments, V applies the input-preserving five-card trick (Section 2) to the two commitments. If the output is 1 (meaning that at least one of the two commitments is white), V continues; otherwise V aborts.

5.3 Security Proofs

Theorem 4 (Completeness) *If P knows a solution of a Hitori grid, then P can convince V.*

Proof Suppose that P knows the solution s of the grid and runs the setup phase. We will demonstrate that P can perform the setup phase correctly, and V never aborts in the verification phase.

Setup: Because s is a solution, the white cells form a white-polyomino. Hence, in Step 3, P can always choose a black commitment to make one of its neighbors white. Hence, in Step 3, V never aborts. For Step 3c, note that the protocol can make a black commitment white, depending on P's will (and V cannot notice it). Let us consider two cases: The first one is that P wants to change the black commitment. The configuration is then:

$$\boxed{♣}\,\boxed{♡}\,.$$
$$\boxed{♣}\,\boxed{♡}$$

Therefore, after applying a pile-shifting shuffle, if the configuration is the same, then the right card of the second row is a heart $\boxed{♡}$; hence, V will swap the cards of the first row, leading to $\boxed{♡}\,\boxed{♣}$. Therefore, the resulting commitment is white. If the configuration changes after applying a pile-shifting shuffle, the resulting configuration is:

$$\boxed{♡}\,\boxed{♣}\,.$$
$$\boxed{♡}\,\boxed{♣}$$

Therefore, V will not change the first row (as the right card of the second row is a club $\boxed{♣}$). The resulting commitment also corresponds to white.

The second case is that P does not want to change the black commitment, and it is analogous to the first case.

Verification: Because s is a solution, each row and each column include at most one occurrence of a given numbered cell. Hence, when the same numbered cell appears k times in a given row or column, at least $k - 1$ commitments among the k ones on the numbered cells are black, according to rule 1. Therefore, if s is a solution, P can always choose a white commitment (if exist) among the k commitments in Step 2, and V never aborts.

The last verification considers the black cells. Because s is a solution, for a given black commitment, its four neighbors are white commitments. Therefore, any combination of pair of these five commitments differs from the pair solely formed by black commitments. Hence, each of the two adjacent commitments differ from the pair formed by two black commitments, which consistently triggers output 1 for the input-preserving five-card trick.   □

Theorem 5 (Soundness) *If P does not provide a solution for a Hitori grid, then P cannot convince V.*

Proof Suppose that P does not provide a solution for the grid. We will demonstrate that V will always detect it. Note that the resulting placement of commitments after executing the setup phase represents a white-polyomino, regardless of P's behavior.

Suppose that the resulting placement does not correspond to a solution. We can distinguish two cases, each corresponding to a rule that is not respected.

- *One occurrence:* For a given row (column), V considers the same numbered cell with $k > 1$ occurrences. If this rule is not respected, it means that more than one white commitment exist among the k commitments on the k numbered cells. V will detect it because $k - 1$ commitments among them are revealed in Step 3.

- *Lonely black:* Note that V checks all possible two-adjacent commitments. Therefore, if P tries to blacken more cells than needed (*e.g.*, for passing the one occurrence verification), then V will detect it.

Hence, if P does not provide a solution, then V will detect it. Finally, if P passes all the verifier's checks, it implies that P possesses the solution, which concludes the soundness proof.   □

Theorem 6 (Zero-knowledge) *V learns nothing about P's solution of a given grid.*

Proof We repeat the same proof technique as in [9]. The simulator acts as follows.

- The simulator constructs a random white-polyomino.

- During the verification of the one occurrence rule, the same numbered cell that appears $k > 1$ times for each row and column is verified. For the k corresponding commitments, the chosen pile protocol is applied; hence, the simulator can swap the $k - 1$ commitments, except the target commitment by black commitments.

- During the verification of the lonely black rule, each pair of adjacent commitments is verified. When the first pile-shifting shuffle is applied in executing the input-preserving five-card trick, the simulator replaces the five cards in the first row with $\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\heartsuit}$ that are randomly shifted.

The simulated and real proofs are indistinguishable. Hence, V learns nothing about P's solution.   □

6 ZKP Protocol for Heyawake

In this section, we present a card-based ZKP protocol for Heyawake. Note that rules 2 and 3 of Heyawake are the same as Hitori. Therefore, we adopt the same setup phase as Hitori (as seen in Section 6.1) and the lonely black verification (as in Section 6.2). Rules 1 and 4 are unique for Heyawake; in Section 6.2, we present their verification phases. We provide the security proofs in Section 6.3.

6.1 Setup Phase

P will be able to create a white polyomino (consisting of white commitments) using the same protocol described in Section 3.

6.2 Verification phase

Lonely black (rule 2)

Because this rule is exactly the same as in Hitori (rule 2), P and V use the input-preserving five-card trick (Section 2) to verify this rule as in our ZKP protocol for Hitori presented in Section 5.

No 3-rooms (rule 4)

V attempts to verify that white cells (white commitments) cannot stretch in straight lines more than two rooms. For this verification, P agrees with V about the cells to verify, following these steps (refer to Figure 4 for examples).

1. For each row, look for a straight line that crosses at least three rooms.
2. Consider all the cells in the middle room, the rightmost cell of the first room, and the leftmost cell of the third room.
3. Repeat the same process for the column, but consider all the cells in the middle room, the lowest cell of the topmost room, and the highest cell of the bottommost room.

For each group of considered cells, P chooses a black commitment using the chosen-pile protocol (Section 2), and V reveals the commitment chosen by P. If the revealed commitment corresponds to black, then V continues; otherwise V aborts. For example, for the group (colored in blue) in the second row in Figure 4, P chooses one commitment among the three commitments; if it is black, V is convinced that white commitments do not cross at these three rooms.

Fig. 4 Two examples of cells (in blue) to consider for rule 4. For clarity, we do not represent all the cells to verify.

Number of black cells in a room (rule 1)

For each room with a numbered cell, V shuffles all the corresponding commitments and reveals them all. If the number of black commitments is equal to the number written on the cell, then V is convinced; otherwise V aborts.

6.3 Security proofs

Theorem 7 (Completeness) *If P possesses a solution of a Heyawake grid, then P can convince V.*

Proof We suppose that P possesses a solution and runs the setup phase. Because this phase is the same as that for Hitori, refer to the proof of Theorem 4 for the connectivity constraint. Next, we check the verification phase.

Verification. Because P possesses the solution, black cells (black commitments) do not vertically nor horizontally contact other black cells. Hence, for each pair of adjacent cells, the five-card trick will always output 1 because the only configuration for obtaining 0 involves two black cells contacting each other.

For the no 3-rooms verification, no straight line of white cells can expand through more than two rooms. Therefore, the cells considered (see Figure 4 for examples) will always comprise at least one black cell.

Because P possesses the solution, the correct number of black cells are placed inside a room. Because the verification only includes a shuffle, the number of black cells remains the same. Hence, the verification continues without aborting, meaning that V is convinced that P has the solution. □

Theorem 8 (Soundness) *If P does not provide a solution of a Heyawake grid , then P cannot convince V.*

Proof Suppose that P does not provide the solution, yet attempts to convince V. We will demonstrate that V will detect it, owing to our protocol. Similar to the soundness proof for Hitori (Theorem 5), we consider that the white cells (white commitments) are connected without loss of generality. We distinguish the cases where the other

rules are not respected. Recall that P does not possess the solution by assumption; hence, at least one of these verifications must fail (otherwise it means that P has the solution).

- *Lonely black:* V checks all possible two-adjacent commitments. Therefore, if P attempts to blacken more cells than needed (*e.g.*, for passing the no 3-rooms verification), then V will detect it.

- *No* 3-*rooms:* V consider all straight lines that cross at least three rooms and reveal one commitment from it. Because P cannot choose a black commitment (if white commitments cross at three rooms), V reveals a white one and aborts the protocol.

- *Black cells in a room:* Each room with a number is inspected; hence, if the rule 1 is not respected, then V will detect it because all the commitments are revealed (after a shuffle).

□

Theorem 9 (Zero-knowledge) *V learns nothing about P's solution of a given grid.*

Proof To prove this theorem, it is sufficient to demonstrate that the revealed values (open cards) follow a distribution that can be simulated without the knowledge of the solution.

Similar to Theorems 6, the connectivity and all the verification, except the no 3-rooms, can be constructed without knowing the solution. For the no 3-rooms phase, any opening card is revealed after a chosen-pile protocol; hence, its position is uniformly distributed among all possible ones.   □

7 Conclusion

In this study, we proposed three card-based ZKP protocols for Nurikabe, Hitori, and Heyawake. These three Nikoli puzzles require that white cells in the solution must be continuous, *i.e.*, form a white-polyomino. We designed the generic method for solving this connectivity constraint challenge. In addition, we developed novel techniques to respect the other rules of the puzzles.

It should be noted that the proposed card-based ZKP protocols are "interactive," *i.e.*, they require a prover P's knowledge or memory to determine how the cards are manipulated during executions of protocols (cf. [5, 15, 21, 29, 31, 32]). Therefore, designing a "non-interactive" protocol for either Nurikabe, Hitori, or Heyawake is an interestingly open problem; "non-interactivity" implies that after P places face-down cards, the players' knowledge or memory is not adopted (as in the standard card-based computation setting [2, 18, 19, 26, 27]).

References

[1] Bultel, X., Dreier, J., Dumas, J., Lafourcade, P.: Physical zero-knowledge proofs for Akari, Takuzu, Kakuro and KenKen. In: Demaine, E.D., Grandoni, F. (eds.) Fun with Algorithms. LIPIcs, vol. 49, pp. 8–1820. Schloss Dagstuhl, Dagstuhl (2016). https://doi.org/10.4230/LIPIcs.FUN.2016.8

[2] Bultel, X., Dreier, J., Dumas, J., Lafourcade, P., Miyahara, D., Mizuki, T., Nagao, A., Sasaki, T., Shinagawa, K., Sone, H.: Physical zero-knowledge proof for Makaro. In: Izumi, T., Kuznetsov, P. (eds.) Stabilization, Safety, and Security of Distributed Systems. LNCS, vol. 11201, pp. 111–125. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03232-6_8

[3] Chien, Y.-F., Hon, W.-K.: Cryptographic and physical zero-knowledge proof: From Sudoku to Nonogram. In: Boldi, P., Gargano, L. (eds.) Fun with Algorithms. LNCS, vol. 6099, pp. 102–112. Springer, Berlin, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13122-6_12

[4] den Boer, B.: More efficient match-making and satisfiability: *The Five Card Trick*. In: Quisquater, J., Vandewalle, J. (eds.) Advances in Cryptology – EUROCRYPT 1989. LNCS, vol. 434, pp. 208–217. Springer, Berlin, Heidelberg (1990). https://doi.org/10.1007/3-540-46885-4_23

[5] Dumas, J., Lafourcade, P., Miyahara, D., Mizuki, T., Sasaki, T., Sone, H.: Interactive physical zero-knowledge proof for Norinori. In: Du, D., Duan, Z., Tian, C. (eds.) Computing and Combinatorics. LNCS, vol. 11653, pp. 166–177. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26176-4_14

[6] Goldreich, O., Kahan, A.: How to construct constant-round zero-knowledge proof systems for NP. J. Cryptology **9**(3), 167–189 (1996). https://doi.org/10.1007/BF00208001

[7] Goldreich, O., Micali, S., Wigderson, A.: Proofs that yield nothing but their validity for all languages in NP have zero-knowledge proof systems. J. ACM **38**(3), 691–729 (1991). https://doi.org/10.1145/116825.116852

[8] Goldwasser, S., Micali, S., Rackoff, C.: Knowledge complexity of interactive proof-systems. Annual ACM Symposium on Theory of Computing, 291–304 (1985). https://doi.org/10.1145/3335741.3335750

[9] Gradwohl, R., Naor, M., Pinkas, B., Rothblum, G.N.: Cryptographic and physical zero-knowledge proof systems for solutions of Sudoku puzzles. Theory Comput. Syst. **44**(2), 245–268 (2009). https://doi.org/10.1007/s00224-008-9119-9

[10] Hearn, R.A., Demaine, E.D.: Games, Puzzles, and Computation. A. K. Peters, Ltd., USA (2009). https://doi.org/10.1201/b10581

[11] Holzer, M., Ruepp, O.: The troubles of interior design–a complexity analysis of the game heyawake. In: Crescenzi, P., Prencipe, G., Pucci, G. (eds.) Fun with Algorithms. LNCS, vol. 4475, pp. 198–212. Springer, Berlin, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72914-3_18

[12] Holzer, M., Klein, A., Kutrib, M., Ruepp, O.: Computational complexity of NURIKABE. Fundam. Informaticae **110**(1-4), 159–174 (2011). https://doi.org/10.3233/FI-2011-534

[13] Isuzugawa, R., Miyahara, D., Mizuki, T.: Zero-knowledge proof protocol for Cryptarithmetic using dihedral cards. In: Kostitsyna, I., Orponen, P. (eds.) Unconventional Computation and Natural Computation. LNCS, vol. 12984, pp. 51–67. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87993-8_4

[14] Koch, A., Walzer, S.: Foundations for actively secure card-based cryptography. In: Farach-Colton, M., Prencipe, G., Uehara, R. (eds.) Fun with Algorithms. LIPIcs, vol. 157, pp. 17–11723. Schloss Dagstuhl, Dagstuhl (2020). https://doi.org/10.4230/LIPIcs.FUN.2021.17

[15] Lafourcade, P., Miyahara, D., Mizuki, T., Robert, L., Sasaki, T., Sone, H.: How to construct physical zero-knowledge proofs for puzzles with a "single loop" condition. Theor. Comput. Sci. **888**, 41–55 (2021). https://doi.org/10.1016/j.tcs.2021.07.019

[16] Miyahara, D., Sasaki, T., Mizuki, T., Sone, H.: Card-based physical zero-knowledge proof for Kakuro. IEICE Trans. Fundamentals **102-A**(9), 1072–1078 (2019). https://doi.org/10.1587/transfun.E102.A.1072

[17] Miyahara, D., Robert, L., Lafourcade, P., Takeshige, S., Mizuki, T., Shinagawa, K., Nagao, A., Sone, H.: Card-based ZKP protocols for Takuzu and Juosan. In: Farach-Colton, M., Prencipe, G., Uehara, R. (eds.) Fun with Algorithms. LIPIcs, vol. 157, pp. 20–12021. Schloss Dagstuhl, Dagstuhl (2020). https://doi.org/10.4230/LIPIcs.FUN.2021.20

[18] Mizuki, T., Shizuya, H.: A formalization of card-based cryptographic protocols via abstract machine. Int. J. Inf. Sec. **13**(1), 15–23 (2014). https://doi.org/10.1007/s10207-013-0219-4

[19] Mizuki, T., Shizuya, H.: Computational model of card-based cryptographic protocols and its applications. IEICE Trans. Fundamentals **100-A**(1), 3–11 (2017). https://doi.org/10.1587/transfun.E100.A.3

[20] Nishimura, A., Hayashi, Y., Mizuki, T., Sone, H.: Pile-shifting scramble for card-based protocols. IEICE Trans. Fundamentals **101-A**(9), 1494–1502 (2018). https://doi.org/10.1587/transfun.E101.A.1494

[21] Ono, H., Manabe, Y.: Card-based cryptographic logical computations using private operations. New Gener. Comput. **39**(1), 19–40 (2021). https://doi.org/10.1007/s00354-020-00113-z

[22] Robert, L., Miyahara, D., Lafourcade, P., Mizuki, T.: Interactive physical ZKP for connectivity: Applications to Nurikabe and Hitori. In: De Mol, L., Weiermann, A., Manea, F., Fernández-Duque, D. (eds.) Connecting with Computability. LNCS, vol. 12813, pp. 373–384. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-80049-9_37

[23] Robert, L., Miyahara, D., Lafourcade, P., Libralesso, L., Mizuki, T.: Physical zero-knowledge proof and NP-completeness proof of Suguru puzzle. Information and Computation, 1–14 (2021). https://doi.org/10.1016/j.ic.2021.104858. in press

[24] Ruangwises, S.: An improved physical ZKP for Nonogram. In: Du, D.-Z., Du, D., Wu, C., Xu, D. (eds.) Combinatorial Optimization and Applications. LNCS, vol. 13135, pp. 262–272. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-92681-6_22

[25] Ruangwises, S.: Two standard decks of playing cards are sufficient for a ZKP for Sudoku. In: Chen, C.-Y., Hon, W.-K., Hung, L.-J., Lee, C.-W. (eds.) Computing and Combinatorics. LNCS, vol. 13025, pp. 631–642. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-89543-3_52

[26] Ruangwises, S., Itoh, T.: Physical zero-knowledge proof for Numberlink puzzle and k vertex-disjoint paths problem. New Gener. Comput. **39**(1), 3–17 (2021). https://doi.org/10.1007/s00354-020-00114-y

[27] Ruangwises, S., Itoh, T.: Physical zero-knowledge proof for Ripple Effect. Theor. Comput. Sci. **895**, 115–123 (2021). https://doi.org/10.1016/j.tcs.2021.09.034

[28] Ruangwises, S., Itoh, T.: Physical ZKP for connected spanning subgraph: Applications to Bridges puzzle and other problems. In: Kostitsyna, I.,

Orponen, P. (eds.) Unconventional Computation and Natural Computation. LNCS, vol. 12984, pp. 149–163. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87993-8_10

[29] Sasaki, T., Miyahara, D., Mizuki, T., Sone, H.: Efficient card-based zero-knowledge proof for Sudoku. Theor. Comput. Sci. **839**, 135–142 (2020). https://doi.org/10.1016/j.tcs.2020.05.036

[30] Shinagawa, K., Mizuki, T., Schuldt, J.C.N., Nuida, K., Kanayama, N., Nishide, T., Hanaoka, G., Okamoto, E.: Card-based protocols using regular polygon cards. IEICE Trans. Fundamentals **100-A**(9), 1900–1909 (2017). https://doi.org/10.1587/transfun.E100.A.1900

[31] Watanabe, Y., Kuroki, Y., Suzuki, S., Koga, Y., Iwamoto, M., Ohta, K.: Card-based majority voting protocols with three inputs using three cards. In: Information Theory and Its Applications, pp. 218–222 (2018). https://doi.org/10.23919/ISITA.2018.8664324

[32] Yasunaga, K.: Practical card-based protocol for three-input majority. IEICE Trans. Fundamentals **E103.A**(11), 1296–1298 (2020). https://doi.org/10.1587/transfun.2020EAL2025