



HAL
open science

Physical ZKP protocols for Nurimisaki and Kurodoko

Léo Robert, Daiki Miyahara, Pascal Lafourcade, Takaaki Mizuki

► **To cite this version:**

Léo Robert, Daiki Miyahara, Pascal Lafourcade, Takaaki Mizuki. Physical ZKP protocols for Nurimisaki and Kurodoko. *Theoretical Computer Science*, 2023, 972, pp.114071. 10.1016/j.tcs.2023.114071 . hal-04434814

HAL Id: hal-04434814

<https://u-picardie.hal.science/hal-04434814>

Submitted on 2 Feb 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Physical ZKP Protocols for Nurimisaki and Kurodoko

Léo Robert^{a,*}, Daiki Miyahara^{b,c}, Pascal Lafourcade^d, Takaaki Mizuki^e

^a*XLIM, University of Limoges, Limoges, France*

^b*University of Electro-Communications, Tokyo, Japan*

^c*National Institute of Advanced Industrial Science and Technology, Tokyo, Japan*

^d*LIMOS, University Clermont-Auvergne, Aubiere, France*

^e*Cyberscience Center, Tohoku University, Sendai, Japan*

Abstract

Proving to someone else the knowledge of a secret without revealing any of its information is an interesting feature in cryptography. The best solution to solve this problem is a *Zero-Knowledge Proof* (ZKP) protocol.

Nurimisaki is a Nikoli puzzle. The goal of this game is to draw a kind of abstract painting (“Nuri”) that represents a sea with some capes (“Misaki”) of an island (represented by white cells). For this, the player has to fulfill cells of a grid in black (representing the sea) in order to draw some capes while respecting some simple rules. One of the specificity of the rules of this game is that every “Misaki” cell can only have one white neighbour.

Kurodoko is also a Nikoli puzzle where some cells need to be blackened in order to ensure that each numbered cell is surrounded by the same number of white cells in the four directions (north, east, south, and west).

Both of these puzzles, Nurimisaki and Kurodoko, share a common connectivity constraint. Using a deck of cards, we propose a physical ZKP protocol for each of the two puzzles above. Our protocols prove that a player knows a solution of (1) a Nurimisaki grid, or (2) a Kurodoko grid, without revealing any information about the solution.

Keywords: Zero-knowledge proof, Pencil Puzzle, Card-based cryptography, Nurimisaki, Kurodoko

*Corresponding author

Email addresses: leo.robert@xlim.fr (Léo Robert), miyahara@uec.ac.jp (Daiki Miyahara), pascal.lafourcade@uca.fr (Pascal Lafourcade), mizuki+lncs@tohoku.ac.jp (Takaaki Mizuki)

1. Introduction

The democratization of computers and network systems has fuelled the virtualization of interactions and processes such as communication, payments, and elections. Proving the knowledge of some secret without revealing any bit of information from that secret is crucial in our today's society. This issue can be applied to numerous contexts.

For instance, a client would like to connect to a server via a password without revealing the password. Another example is database management, where an entity could ask if a piece of information is in a database without asking for factual data. A third example can be given in the electronic voting system where voters want to be sure that ballots are correctly mixed (without revealing how the mix was done). Finally, crypto-currencies, such as Bitcoin, Monero, or Zcash, are eager to include a mechanism to enforce knowledge of some secrets without revealing it (*e.g.*, for anonymous transactions).

A cryptographic tool exists for all the previous examples, called a *Zero-Knowledge Proof* (ZKP) protocol. It enables a prover P to convince a verifier V that P knows a secret s without revealing anything other than it. A ZKP protocol must satisfy the following three properties:

- **Completeness:** If P knows s then the protocol ends without aborting (meaning that V is convinced that P has s);
- **Soundness:** If P does not have s then V will detect it;
- **Zero-Knowledge:** The information given during the protocol leads to V learning nothing about s .

In practice, ZKP protocols are typically executed by computers. However, we present here two protocols using physical objects like playing cards and envelopes. This line of research has been growing a lot in recent years and our aim is to extend this field.

The first physical ZKP protocol [1] for a Sudoku grid was constructed using a deck of cards. Since this novel protocol was devised, several teams in the world have proposed physical ZKP protocols using a deck of cards for pencil puzzles, such as Sudoku [2, 3], Akari [4], Takuzu [4], Kakuro [4, 5], KenKen [4], Makaro [6], Norinori [7], Slitherlink [8], Suguru [9, 10], Nurikabe [11], Ripple Effect [12], Numberlink [13], Bridges [14], Cryptarithmic [15], Shikaku [16], Usowan [17], and Nonogram [18, 19].

In this paper, we propose ZKP protocols for two other Nikoli’s puzzle, Nurimisaki and Kurodoko. Why shall we propose a new card-based ZKP protocol for another Nikoli puzzle? For us, it is similar to the question: Why shall we prove that a puzzle is NP-complete? People want to know if a puzzle is NP-complete in order to know if the puzzle is difficult or not for a computer to solve it. Card-based ZKP protocols are quite similar; once a puzzle is shown to be NP-complete, a natural question is: Can we design an efficient physical ZKP protocol? This is an intellectual challenge on those puzzles.

Moreover, each puzzle has different rules and specificity, which force us to imagine new physical ZKP techniques. For instance, consider Nurimisaki, which we will deal with in this paper; then, its rules combine for the first time some connectivity, neighbourhood restriction, and straight line with counting, as seen later. A previous work [20] (in Japanese, unpublished) proposed a card-based ZKP protocol for Nurimisaki. Yet, the protocol is not optimal since it prepares another grid to verify the rules. Moreover, elaborate but complex techniques are used (*e.g.*, using another grid to represent the in-spanning-tree of P ’s solution). In contrast, our protocol has a more direct approach with closer interaction to the real game. As for Kurodoko, we will also construct a card-based ZKP protocol with a direct approach. Note that no previous work on ZKP protocols for Kurodoko has been proposed.

Before giving our contributions, let us define the rules of the Nurimisaki and Kurodoko puzzles. Notice that both of these puzzles share a common rule, *i.e.*, some cells must be connected.

Nurimisaki Rules. Figure 1 shows a puzzle instance of Nurimisaki¹. The goal is to color in black some cells on the grid, under the following rules:

1. White cells are connected.
2. A cell with a circle is called a *Misaki*. A Misaki has only one cell of its neighbours (vertically or horizontally) remaining white and the rest black.
3. The number written in a Misaki cell indicates the number of white cells in straight line from the Misaki. If there is no number, any number of white cells is allowed.

¹Example taken from: <https://www.nikoli.co.jp/en/puzzles/nurimisaki/>

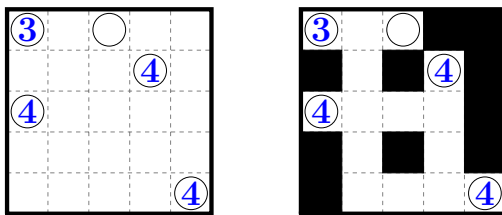


Figure 1: Nurimisaki example (left) with its solution (right).

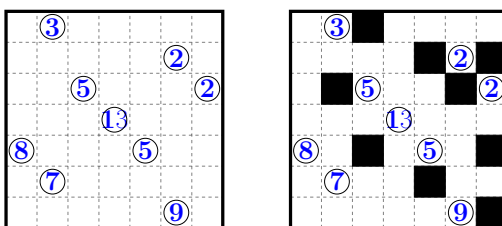


Figure 2: Kurodoko example (left) with its solution (right).

4. Any white cell without a circle cannot be a Misaki.
5. A 2×2 square cannot be composed of only black or white cells.

Kurodoko rules. The goal of this puzzle, as illustrated in Fig. 2², is to fill some cells in black, with the following rules:

1. White cells are connected.
2. Black cells cannot touch vertically nor horizontally.
3. Cells with number remain white.
4. The number on a cell indicates the number of white cells from the numbered cell to the edge of the grid or a black cell (in the four directions).

Nurimisaki puzzle was recently proven NP-complete in [21] and Kurodoko in [22]; hence, it is a natural question to construct physical ZKP protocols for these fun puzzles. Although Goldwasser *et al.* [23] proved that any NP-complete problem has its corresponding interactive ZKP protocol, simple physical ZKP protocols are always solicited as mentioned above.

²Example taken from: <https://www.nikoli.co.jp/en/puzzles/kurodoko/>

Contributions. For each of Nurimisaki and Kurodoko, we propose a physical ZKP protocol that only uses cards and envelopes. We rely on some classical existing card-based sub-protocols in order to be able to construct our ZKP protocols.

The main difficulty in Nurimisaki that seems to be simple, is that the existing techniques proposed in the literature for few years cannot be applied directly. The main trick is to find an encoding that allows us to apply several sub-protocols in the right order to obtain a secure ZKP protocol.

For Kurodoko, the problem lies on how to count some commitments on heterogeneous sequences. This will be solved with a new sub-protocol used in such a way that enumeration is correct.

Overall, we propose original ways to combine several techniques to design our ZKP protocols with a reasonable amount of cards and manipulations.

Note that this work is an extended version of a previously published conference paper [24] where only the Nurimisaki protocol was proposed. That is, Sects. 5 and 6 are totally new materials.

Outline. In Sect. 2, we introduce our encoding scheme using cards in order to represent a grid of the game and a solution. We also give some sub-protocols that are used in our construction. In Sect. 3, we give our ZKP protocol for Nurimisaki, and the security proofs in Sect. 4. Then, a Kurodoko protocol is proposed in Sect. 5 along with security proofs in Sect. 6 before concluding in Sect. 7.

2. Preliminaries

We explain the notations and sub-protocols used in our constructions.

Cards and Encoding. The cards we use in our protocols consist of clubs $\clubsuit\clubsuit \dots$, hearts $\heartsuit\heartsuit \dots$, and numbered cards $\boxed{1}\boxed{2} \dots$, whose backs are identical $\boxed{?}$. We encode three colors {black, white, red} with the order of two cards as follows:

$$\clubsuit\heartsuit \rightarrow \text{black}, \quad \heartsuit\clubsuit \rightarrow \text{white}, \quad \heartsuit\heartsuit \rightarrow \text{red}. \quad (1)$$

We call a pair of face-down cards $\boxed{?}\boxed{?}$ corresponding to a color according to the above encoding rule a *commitment* to the respective color. We also use the terms, a *black commitment*, a *white commitment*, and a *red commitment*. We sometimes regard black and white commitments as bit

values, based on the following encoding scheme:

$$\boxed{\heartsuit}\boxed{\clubsuit} \rightarrow 0, \quad \boxed{\clubsuit}\boxed{\heartsuit} \rightarrow 1. \quad (2)$$

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 $\underbrace{\boxed{?}\boxed{?}}_x$.

We also define two other encoding [25, 12]:

- **\clubsuit -scheme**: for $x \in \mathbb{Z}/p\mathbb{Z}$, there are p cards composed of $p - 1$ \heartsuit s and one \clubsuit , where the \clubsuit is located at position $(x + 1)$ from the left. For example, 2 in $\mathbb{Z}/4\mathbb{Z}$ is represented as $\boxed{\heartsuit}\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\heartsuit}$.
- **\heartsuit -scheme**: it is the same encoding as above but the \heartsuit and \clubsuit are reversed. For instance, 2 in $\mathbb{Z}/4\mathbb{Z}$ is represented as $\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\heartsuit}\boxed{\clubsuit}$.

2.1. Pile-shifting Shuffle

This shuffling action from [25, 26], 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 $(\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_m)$, applying a *pile-shifting shuffle* (denoted by $\langle \cdot \| \dots \| \cdot \rangle$) results in $(\mathbf{p}_{s+1}, \mathbf{p}_{s+2}, \dots, \mathbf{p}_{s+m})$:

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

where s is uniformly and randomly chosen from $\mathbb{Z}/m\mathbb{Z}$. Implementing a pile-shifting shuffle is simple: we use physical cases that can store a pile of cards, such as boxes and envelopes; a player (or players) cyclically shuffles them manually until everyone (*i.e.*, the prover P and the verifier V) loses track of the offset.

2.2. Input-preserving Five-card Trick

Given two commitments to $a, b \in \{0, 1\}$ based on the encoding rule (2), this sub-protocol [27, 28] reveals only the value of $a \vee b$ as well as restores commitments to a and b : $\underbrace{\boxed{?}\boxed{?}}_a \underbrace{\boxed{?}\boxed{?}}_b \rightarrow a \vee b \ \& \ \underbrace{\boxed{?}\boxed{?}}_a \underbrace{\boxed{?}\boxed{?}}_b$.

1. Add helping cards and swap the two cards of the commitment to b so that we have the negation \bar{b} , as follows:

$$\underbrace{\boxed{?}\boxed{?}}_a \underbrace{\boxed{?}\boxed{?}}_b \rightarrow \underbrace{\boxed{?}\boxed{?}}_a \heartsuit \underbrace{\boxed{?}\boxed{?}}_{\bar{b}} \heartsuit \clubsuit \clubsuit \clubsuit \clubsuit.$$

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

$$\boxed{?}\boxed{?}\heartsuit\boxed{?}\boxed{?}\heartsuit\clubsuit\clubsuit\clubsuit\clubsuit \rightarrow \begin{array}{|c|c|c|c|c|} \hline \boxed{?} & \boxed{?} & \heartsuit & \boxed{?} & \boxed{?} \\ \hline \heartsuit & \clubsuit & \clubsuit & \clubsuit & \clubsuit \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|c|} \hline \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ \hline \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ \hline \end{array}.$$

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

$$\left\langle \begin{array}{|c|} \hline \boxed{?} \\ \hline \end{array} \parallel \begin{array}{|c|} \hline \boxed{?} \\ \hline \end{array} \parallel \begin{array}{|c|} \hline \boxed{?} \\ \hline \end{array} \parallel \begin{array}{|c|} \hline \boxed{?} \\ \hline \end{array} \parallel \begin{array}{|c|} \hline \boxed{?} \\ \hline \end{array} \right\rangle \rightarrow \begin{array}{|c|c|c|c|c|} \hline \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ \hline \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ \hline \end{array}.$$

4. Reveal all the cards in the first row.
 - (a) If it is $\clubsuit\clubsuit\heartsuit\heartsuit\heartsuit$ (up to cyclic shifts), then $a \vee b = 0$.
 - (b) If it is $\heartsuit\clubsuit\heartsuit\clubsuit\heartsuit$ (up to cyclic shifts), then $a \vee b = 1$.
5. After turning over all the face-up cards, apply a pile-shifting shuffle.
6. Reveal all the cards in the second row; then, the revealed cards should include exactly one \heartsuit .
7. Shift the sequence of piles so that the revealed \heartsuit is the leftmost card and swap the two cards of the commitment to \bar{b} to restore commitments to a and b .

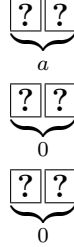
Note that we can also compute an AND operation using De Morgan's laws: $a \wedge b = \overline{\overline{a} \vee \overline{b}}$.

2.3. Mizuki–Sone Copy Protocol

Given a commitment to $a \in \{0, 1\}$ along with four cards $\clubsuit\heartsuit\clubsuit\heartsuit$, the Mizuki–Sone copy protocol [29] outputs two commitments to a :

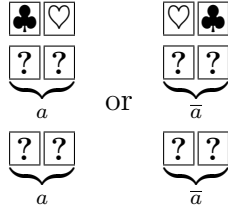
$$\underbrace{\boxed{?}\boxed{?}}_a \clubsuit\heartsuit\clubsuit\heartsuit \rightarrow \underbrace{\boxed{?}\boxed{?}}_a \underbrace{\boxed{?}\boxed{?}}_a.$$

1. Turn all cards face-down and set the commitments as follows:



2. Apply a pile-shifting shuffle as follows: $\left\langle \begin{array}{|c|} \hline \boxed{?} \\ \hline \boxed{?} \\ \hline \boxed{?} \\ \hline \end{array} \parallel \begin{array}{|c|} \hline \boxed{?} \\ \hline \boxed{?} \\ \hline \boxed{?} \\ \hline \end{array} \right\rangle \rightarrow \begin{array}{|c|c|} \hline \boxed{?} & \boxed{?} \\ \hline \boxed{?} & \boxed{?} \\ \hline \boxed{?} & \boxed{?} \\ \hline \end{array}.$

3. Reveal the two above cards to obtain either a or \bar{a} as follows:



2.4. How to Form a White Polyomino

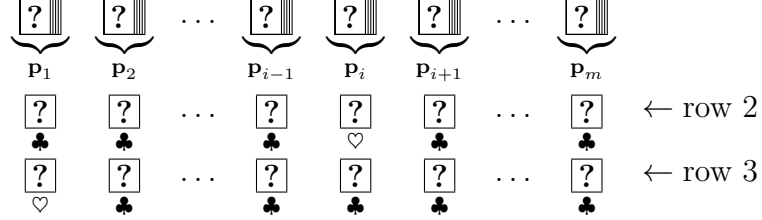
We introduce the generic method of [11] to address the connectivity constraint. Given a grid where all cells are black, it enables a prover P to make white connected cells, *i.e.*, a *white-polyomino* (*i.e.*, a shape composed of any number of connected white cells), without revealing anything to a verifier V . We first describe two crucial sub-protocols: the chosen pile protocol and the 4-neighbor protocol.

Chosen pile protocol [7]. The chosen pile protocol allows P to choose a pile of cards without V knowing which one it is. This pile can be manipulated and all the commitments are replaced to their initial order afterward.

This protocol is an extended version of the “chosen pile cut” proposed in [30]. Given m piles $(\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_m)$ with $2m$ additional cards, the *chosen pile protocol* enables a prover P to choose the i -th pile \mathbf{p}_i (without revealing the index i) and revert the sequence of m piles to their original order after applying other operations to p_i .

1. Using $m - 1$ \clubsuit s and one \heartsuit , P places m face-down cards encoding $i - 1$ in the \heartsuit -scheme (denoted by *row 2*) below the given piles, *i.e.*,

only the i -th card is \heartsuit . We further put m cards encoding 0 in the \heartsuit -scheme (denoted by *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*. Then, exactly one \heartsuit appears, and the pile above the revealed \heartsuit is the i -th pile (and 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 placed back to the i -th position in the sequence.
4. Remove the revealed cards, *i.e.*, the cards in *row 2*. (Note, therefore, that we do not use the card \heartsuit revealed in Step 3.) Then, apply a pile-shifting shuffle.
5. Reveal all the cards in *row 3*. Then, 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.

Sub-protocol: 4-Neighbor Protocol [11]. Given pq commitments placed on a $p \times q$ grid, a prover P has a commitment in mind, which we call a *target* commitment. The prover P wants 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 it is) as well as V should be able to confirm the colors of both the commitments. To handle the case where the target commitment is at the edge of the grid, we place commitments to red (as “dummy” commitments) in the left of the first column and below the last row to prevent P from choosing a commitment that is not a neighbor. Thus, the size of the expanded grid is $(p + 1) \times (q + 1)$. This sub-protocol 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:
 $\boxed{??} \boxed{??} \boxed{??} \boxed{??} \cdots \boxed{??}$.
2. P uses the chosen pile protocol to reveal the target commitment.
3. P and V pick all the four neighbors of the target commitment. Since a pile-shifting shuffle is a cyclic reordering, the distance between commitments are kept (up to a given modulo). That is, for a target commitment (not at any the edge), the possible four neighbors are at distance one for the left or right one, and $p + 1$ for the bottom or top one so that P and V can determine the positions of all the four neighbors.
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.

Forming white-polyomino. Assume that there is a grid having $p \times q$ cells. P wants to arrange white commitments on the grid such that they form a white-polyomino while V is convinced that the placement of commitments is surely a white-polyomino. The sub-protocol proceeds 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 above so 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 wants to change into a white one.
 - (a) V swaps the two cards constituting the chosen commitment so 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.

Notice that this step is optional for Nurimisaki and Kurdoko since white commitments can be known to V given the rules of each game.

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 wants 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 keep it black (depending on P 's choice) by executing the following steps; otherwise V aborts.
 - i. If P wants to change the commitment, P places face-down club-to-heart pair below it; otherwise, P places a heart-to-club pair:

$$\boxed{?}\boxed{?} \rightarrow \begin{array}{cc} \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} \\ \clubsuit & \heartsuit \end{array} \quad \text{or} \quad \begin{array}{cc} \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} \\ \heartsuit & \clubsuit \end{array}.$$

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

$$\left\langle \begin{array}{c} \boxed{?} \\ \boxed{?} \end{array} \parallel \begin{array}{c} \boxed{?} \\ \boxed{?} \end{array} \right\rangle \rightarrow \begin{array}{cc} \boxed{?} & \boxed{?} \\ \boxed{?} & \boxed{?} \end{array}.$$

- iii. V reveals the two cards in the second row. If the revealed right card is \heartsuit , then V swaps the two cards in the first row; otherwise V does nothing.
- (d) P and V end the 4-neighbor protocol.

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

2.5. Sum in \mathbb{Z}

We describe the protocol developed in [12] for adding elements in $\mathbb{Z}/2\mathbb{Z}$ with a result in \mathbb{Z} .

Given n commitments $x_1, x_2, \dots, x_n \in \mathbb{Z}/2\mathbb{Z}$ along with one \clubsuit and \heartsuit , the protocol produces their sum $S = \sum_{i=1}^n x_i$ in $\mathbb{Z}/(n+1)\mathbb{Z}$ encoded in the \heartsuit -scheme without revealing the value of x_i for every $i \in \{1, \dots, n\}$.

The idea is to compute the sum inductively; when starting by the two first commitments to x_1 and x_2 , they are transformed into $x_1 - r$ and $x_2 +$

r encoded in the \heartsuit -scheme and \clubsuit -scheme, respectively, for a uniformly random value $r \in \mathbb{Z}/3\mathbb{Z}$. Then $x_2 + r$ is revealed (no information about x_2 is revealed because r is random), and $x_1 - r$ is shifted by $x_2 + r$ positions to encode $(x_1 - r) + (x_2 + r) = x_1 + x_2$. Note that this result is in $\mathbb{Z}/(n+1)\mathbb{Z}$ (or simply \mathbb{Z} because the result is less than or equal to p) for elements x_1, x_2 in $\mathbb{Z}/n\mathbb{Z}$.

The protocol is now formally described. First notice that a black commitment is assumed to be equal to 1 and a white commitment is equal to 0 (according to Eqs. (1) and (2)). Consider first two commitments to x_1 and x_2 (either 0 or 1):

$$\underbrace{\boxed{?}\boxed{?}}_{x_1} \underbrace{\boxed{?}\boxed{?}}_{x_2} \clubsuit \heartsuit \rightarrow \underbrace{\boxed{?}\boxed{?}\boxed{?}}_{x_1+x_2}.$$

1. Swap the two cards of the commitment to x_1 and add a \clubsuit face down to the right. Those three cards represent x_1 in the \heartsuit -scheme in $\mathbb{Z}/3\mathbb{Z}$:

$$\overleftrightarrow{\underbrace{\boxed{?}\boxed{?}}_{x_1}} \boxed{?} \clubsuit \rightarrow \underbrace{\boxed{?}\boxed{?}\boxed{?}}_{x_1}.$$

2. Add a \heartsuit on the right of the commitment to x_2 . Those three cards represent x_2 in the \clubsuit -scheme in $\mathbb{Z}/3\mathbb{Z}$:

$$\underbrace{\boxed{?}\boxed{?}}_{x_2} \boxed{?} \heartsuit \rightarrow \underbrace{\boxed{?}\boxed{?}\boxed{?}}_{x_2}.$$

3. Obtain three cards representing $x_1 + r$ and those representing $x_2 - r$ for a uniformly random value $r \in \mathbb{Z}/3\mathbb{Z}$ as follows.

- (a) Place in *reverse* order the three cards obtained in Step 2 below the three cards obtained in Step 1:

$$\underbrace{\boxed{?}\boxed{?}\boxed{?}}_{x_1} \underbrace{\boxed{?}\boxed{?}\boxed{?}}_{x_2} \rightarrow \begin{array}{c} \underbrace{\boxed{?}\boxed{?}\boxed{?}}_{x_1} \\ \underbrace{\boxed{?}\boxed{?}\boxed{?}}_{2-x_2} \end{array}.$$

(b) Apply a pile shifting shuffle as follows:

$$\left\langle \begin{array}{|c|} \hline ? \\ \hline ? \\ \hline \end{array} \parallel \begin{array}{|c|} \hline ? \\ \hline ? \\ \hline \end{array} \parallel \begin{array}{|c|} \hline ? \\ \hline ? \\ \hline \end{array} \right\rangle \rightarrow \begin{array}{c} \underbrace{\boxed{?} \boxed{?} \boxed{?}}_{x_1+r} \\ \underbrace{\boxed{?} \boxed{?} \boxed{?}}_{2-x_2+r} \end{array} .$$

For a uniformly random value $r \in \mathbb{Z}/3\mathbb{Z}$, we obtain three cards representing $x_1 + r$ and those representing $2 - x_2 + r$.

(c) Rearrange the three cards representing $2 - x_2 + r$ to obtain those representing $x_2 - r$:

$$\underbrace{\boxed{?} \boxed{?} \boxed{?}}_{x_1+r} \quad \underbrace{\boxed{?} \boxed{?} \boxed{?}}_{x_2-r} .$$

4. Reveal the three cards representing $x_2 - r$, and shift to the right the three cards representing $x_1 + r$ to obtain those representing $x_1 + x_2$ in the \heartsuit -scheme; apply the same routine for the remaining elements to compute the final sum.

Notice that we described the protocol for a result in $\mathbb{Z}/3\mathbb{Z}$ but it is easily adaptable for a result in, let say, $\mathbb{Z}/q\mathbb{Z}$.

3. ZKP Protocol for Nurimisaki

In this section, we present our ZKP protocol for Nurimisaki. Hereinafter, we consider an instance of Nurimisaki as a rectangular grid of size $p \times q$.

3.1. Setup phase

The verifier V and the prover P place black commitments on all the cell of the $p \times q$ grid and place red commitments (“dummy” commitments) around the grid so that we have $(p + 1)(q + 1)$ commitments.

3.2. Connectivity phase

P and V change the commitments of one Misaki cell to white (chosen arbitrarily). This allows to remove the Step 2 of *Forming white-polyomino* in Sect. 2.4. The remaining protocol is then processed, a white-polyomino is formed according to P 's solution. Now, V reveals all the commitments corresponding to Misaki to check that they are indeed white. After this phase, V is convinced that white commitments are connected (rule 1).

3.3. Verification Phase

The verifier V is now checking that the other rules are satisfied.

No 2×2 square (rule 5). We use an adapted verification phase of the one in [11] for checking that 2×2 squares are not composed of only white (black) commitments. Note that for an initial grid $p \times q$, there are $(p - 1)(q - 1)$ possible squares of size 2×2 . Thus P and V consider each of those squares (in any order) and apply the following:

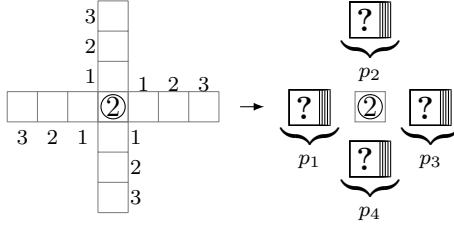
1. P chooses a white commitment using the chosen-pile protocol (Sect. 2.4), and reveals it. After putting back the commitments in their initial position, P chooses a black commitment with the chosen-pile protocol, and reveals it³.
2. If there are exactly a white commitment and a black commitment revealed at the previous step, V continues; otherwise, abort.

Misaki (rule 2 and 3). V wants to check that each Misaki cell (which is a cell with a circle) has only one of its neighbours white and others black. Moreover, when a Misaki has a number in it, V wants to check that the straight line formed by white cells starting from the Misaki cell has the corresponding number of white cells.

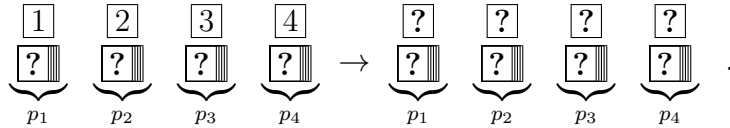
P and V first consider Misaki cells with a number. For each Misaki cell (not at a border) with a number i on it, apply the following:

1. P and V add black commitments (*i.e.*, “dummy” commitments) at the border of the grid. This ensures that we delimit correctly the number of white commitments in a straight line.
2. For each of the four neighbours, P and V form a pile composed of $i + 1$ commitments for each direction. Let p_1 , p_2 , p_3 , and p_4 denote the four piles. We show an example where $i = 2$ as follows.

³We apply twice the chosen-pile protocol to avoid leak. Indeed, if the commitments are not adjacent (*i.e.*, in diagonal), the pile-shifting shuffle of the chosen-pile protocol keeps the order of commitments. So non-adjacent commitments will be noticed after the pile-shifting shuffle.



3. P and V place numbered cards above the piles and turn them face-down as follows:



4. P and V shuffle the piles and reveal the first commitment of each pile. If there is exactly one commitment corresponding to white then V continues. Otherwise, V aborts.
5. V reveals the next i commitments of the pile with the first white commitment. If there are only white commitments for the first $i - 1$ commitments and a black commitment for the last one, then V continues; otherwise, aborts.
6. P and V turn all the revealed cards face-down and shuffle the piles again.
7. P and V reveal the four numbered cards to know which piles are p_1 , p_2 , p_3 , and p_4 . Then, they place the four piles back to their original positions.

After this step, V is convinced that Misaki cells with a number are well-formed. In the case where there is no number, the first step consists of forming a pile with only one commitment. Hence, V is convinced that Misaki cells without a number satisfy only rule 2 but not rule 3 since any number of white cells could form the straight line.⁴

⁴Note that we described the protocol for Misaki cell not at the border of the grid. If a Misaki cell is at a border (but not a corner) then the 4-neighbours becomes the 3-neighbours and the protocol is the same (there will be only three piles instead of four). For Misaki cells at a corner, P and V consider the 2-neighbours (thus only two piles).

No circle, no Misaki (rule 4). V needs to check that every white cell without a circle is not a Misaki, meaning that any white cell of the grid has at least two of its neighbours white. This rule is somewhat challenging to verify without leaking information on the solution because the number and location of white cells are part of the solution (and must not be publicly revealed).

If the targeted cell is black then there is nothing to verify since any configurations could occur. Yet, if the targeted cell is white then there are at least (but it could be more) two neighbours that are white. The idea is to set the value of targeted cell being 4 if it is white and 0 if it is black. Then we add the neighbours to it (white is 0, and black is 1). If the cell is black then the sum is always less than or equal to 4 (which is permitted by the rules to have all black). But if the cell is white then the permitted value for the sum is less than or equal to 6 (a Misaki is equal to 7) for a targeted cell that is not at a border.

For a given cell, called a targeted cell c_t , we look at its neighbors (up to 4). The idea of verifying that a white cell is not a Misaki is to first sum the four neighbors (where a white cell is equal to 0 and a black cell is 1). Then by choosing another encoding, the targeted cell can be equal to 4 for white and 0 for black. Finally, adding the sum of the neighbors with c_t gives at most 4 for black c_t (which is permitted by the rules) and at most 6 for white c_t in a valid configuration and 7 or 8 for invalid configuration.

1. Copy all the commitments using the copy protocol (Sect. 2.3). The number of copies for a $p \times q$ grid is $2(2pq - p - q)$; we leave the detailed computations in [Appendix A](#).
2. Sum the four neighbours by considering that a white commitment is equal to 0 and a black commitment is equal to 1. The result is given in the \heartsuit -scheme (*i.e.*, there are four \clubsuit s and one \heartsuit at position corresponding to the result of the sum).
3. For the targeted cell, add three \heartsuit s in the middle of the commitment as:

$$\begin{aligned} \text{white: } \heartsuit \clubsuit &\rightarrow \heartsuit \heartsuit \heartsuit \heartsuit \clubsuit = 4, \\ \text{black: } \clubsuit \heartsuit &\rightarrow \clubsuit \heartsuit \heartsuit \heartsuit \heartsuit = 0. \end{aligned}$$

White is now 4 and black is 0 in the \clubsuit -scheme.

4. Sum the result of the two previous steps (the sum of the four neighbours and the inner cell). The result is encoded in the \heartsuit -scheme.
5. Reveal the last and penultimate cards. If a \heartsuit appears then abort; otherwise, continue.

4. Security Proofs for Nurimisaki

Our protocol needs to verify three security properties given as theorems. In all theorems, we use the standard physical security assumptions but only implicitly. Indeed, we assume that:

- cards are indistinguishable when face down (represented as \heartsuit);
- envelopes enforce non-malleability of commitments, which ensure the integrity of the cards and their order in sequence;
- shuffles are assumed to be *perfect* in the sense that both parties (P and V) lose track of the initial sequence.
- the last two assumptions are composable, meaning that shuffling envelopes guarantees the integrity of the commitments (inside envelopes) and outputs a sequence with indices re-labeled as a random permutation.

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

PROOF. First, notice that P convinces V in the sense that the protocol does not abort, which means that all the rules are satisfied. The protocol can be split in two: (1) the connectivity and (2) the verification phases.

(1) Since P knows the solution, the white cells are connected and hence P can always choose a black commitment at step 2 to swap it to white. Notice that there exists a proof for the connectivity in [11].

(2) The verification of 2×2 square will not abort since if P has the solution then for any given 2×2 square there always exist a white commitment and a black commitment. For the Misaki rule, each Misaki cell has three of its neighbors black and one white; thus, the first commitment of piles p_1, p_2, p_3, p_4 will reveal exactly three black and one white commitments. Then, when looking at pile p_j of the first commitment corresponding

to white, the number of white commitments corresponds to the number in the inner cell. Thus the protocol will continue. Finally, the non-Misaki rule is verified. Since P has the solution, any white cell (with no circle in it) has at least two white neighbors. Thus if the inner cell is white then the sum will start to 4 and the maximal value is 6 because a solution has at least two whites so at most two black commitments (of value 1 in this step). Therefore, the protocol will continue and hence V will be convinced that P has the solution. \square


Theorem 2 (Soundness). *If P does not provide a solution of the $p \times q$ Nurimisaki grid, P is not able to convince V .*

PROOF. Suppose that P does not know the solution; then, at least one of the rules is not verified. If the white cells are not connected then P cannot choose a black commitment at step 2, and hence V will detect it. Notice that there is also the proof of this phase in [11].

If P does not have the solution, then one of the verification phases will fail. We apply a case distinction for those verifications. Assume first that there is a block of 2×2 square composed of only white (black) commitments, then P cannot choose, during the chosen-pile protocol, two distinct commitments (*i.e.*, a black and a white) thus the revealed commitments will attest to V that P does not have the solution. Second, assume that a Misaki cell is not well-formed in the sense that either (1) the number of white neighbours is not equal to 1 or that (2) the number of white cells in straight line does not correspond to the number of Misaki cells. For (1) the neighbours are revealed (after a shuffle) so V will notice the number of white commitments; for (2) all the commitments next to the white neighbour are revealed thus V will also notice if there is a flaw. The last verification is for white cells which are not Misaki. It is equivalent of saying that any white cell (without a circle in it) has at least two white neighbours. If a white cell has only one white neighbour then during the sum process, then $c_t = 4$ (because the central cell is white) and the total for its neighbours is 3 (because there are three black commitments and one white). The final sum is then equal to 7, since V will look at the last and penultimate card of the sum (corresponding to a sum equal to 8 and 7) then V will detect that a white card is a Misaki. Notice that a sum equals to 8 means the white cell is surrounded by four black cells. It is not possible since white cells are connected. \square

Theorem 3 (Zero-knowledge). *V learns nothing about P’s solution of the given grid G.*

PROOF. We use the same proof technique as in [1], namely the description of an efficient *simulator* which simulates the interaction between an honest prover and a cheating verifier. The goal is to produce an indistinguishable interaction from the verifier’s view (with the prover). Notice that the simulator does not have the solution but it can swap cards during shuffles as used in [1].

Informally, the verifier cannot distinguish between two protocols, one that is run with the actual solution and one with random commitments. The simulator acts as follows: The simulator constructs a random connected white polyomino. During the 2×2 square verification, the simulator will swap cards to choose white and black commitments. For the Misaki verification, the simulator swaps three commitments to black for three piles and one to white for the last pile. The latter will also be modified by the simulator to contain the correct numbers of white commitments (and the last commitment to black). During the non-Misaki verification, when the sum is computed, the simulator swaps the cards to always put  cards in position 7 and 8 (for the cell not at the edge, but the latter is done the same way).

The simulated and real proofs are indistinguishable, and hence, *V* learns nothing from the connectivity and verification phases. Finally, we conclude that the protocol is zero-knowledge. \square

5. ZKP Protocol for Kurodoko

In this section, we construct a ZKP protocol for Kurodoko. Before describing our protocol, we first give a roadmap explaining the high-level steps in Fig. 3. We consider a $p \times q$ rectangular grid.

The most challenging part is the verification of rule 4 because the number of white cells from a numbered cell for each direction must be secret (although the total number is given as public information). Ruangwises and Itoh [31] also argued that the verification of the mathematical meaning of numbers is challenging if they must be secret.

Our idea is depicted in Fig. 4: we consider each direction of a given numbered cell (north, east, south, and east) and keep white commitments until reaching a black commitment; then next commitments are turned black (reading from the numbered cell to the edge). The last step is to

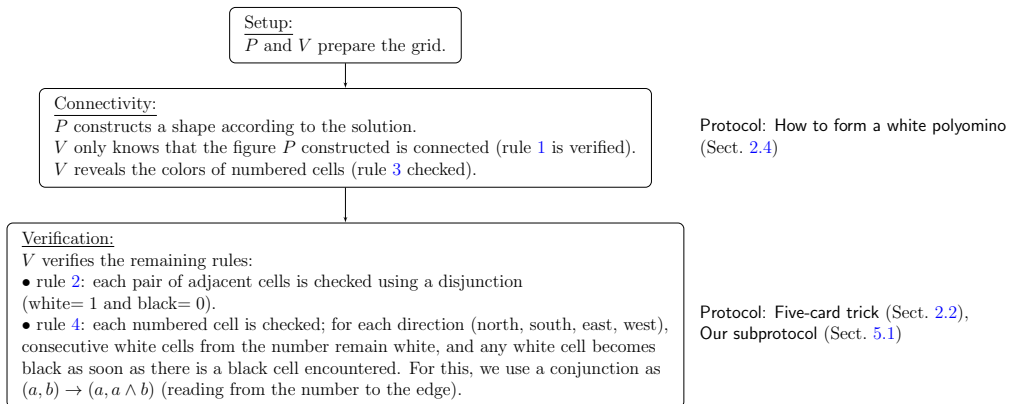


Figure 3: Overview of our protocol for Kurodoko.

shuffle all the commitments in the four directions and reveal to check if the number of white commitments corresponds to the numbered cell. Note that since the operations in the verification process is destructive *i.e.*, we cannot put back the commitments in their original state, we need to copy commitments sharing numbered cell's directions.

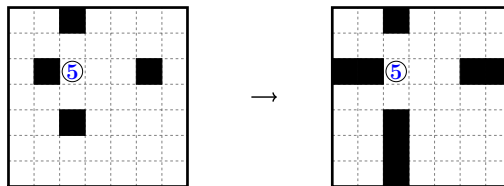


Figure 4: Overview of the preparation for verifying the numbered cell rule.

So, reading from the numbered cell to an edge, we apply an AND operator on each two consecutive cells where black means a value of 0 and white means 1, as depicted in Fig. 5. We show a subprotocol to achieve this in Sect. 5.1.

5.1. Subprotocol

Remember that we want to obtain $(a, a \wedge b)$ from $(a, b) \in \{0, 1\}^2$ while keeping inputs. Given commitments to $a, b \in \{0, 1\}$, we propose a new subprotocol that outputs commitments to a and $a \wedge b$ while keeping input commitments to a and b . Our subprotocol employs the existing protocol presented by Nishida *et al.* [32], which outputs commitments to $a \wedge b$ and

(a, b)		$(a, a \wedge b)$	
1	0	1	$1 \wedge 0$
0	1	0	$0 \wedge 1$
1	1	1	$1 \wedge 1$
0	0	0	$0 \wedge 0$

Figure 5: Changing the consecutive cells from left to right where $a, b \in \{0, 1\}$ represents the colors of the left and right cells, respectively.

b . To output two commitments to a , we employ the Mizuki–Sone copy protocol [29] introduced in Sect. 2.3 and combine it with Nishida *et al.*'s protocol [32]. In other words, our subprotocol executes both protocols simultaneously.

Given commitments to a and b along with six cards, our subprotocol outputs a commitment to $a \wedge b$, two commitments to a , and a commitment to b :

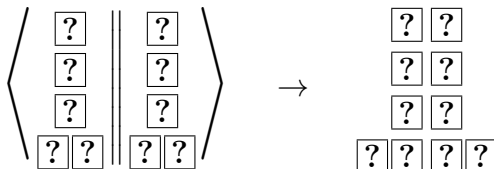
$$\underbrace{\boxed{?} \boxed{?} \boxed{?} \boxed{?}}_a \underbrace{\clubsuit \heartsuit \clubsuit \heartsuit \clubsuit \heartsuit}_b \rightarrow \underbrace{\boxed{?} \boxed{?} \boxed{?} \boxed{?}}_{a \wedge b} \underbrace{\boxed{?} \boxed{?}}_a \underbrace{\boxed{?} \boxed{?}}_a \underbrace{\boxed{?} \boxed{?}}_b .$$

It proceeds as follows.

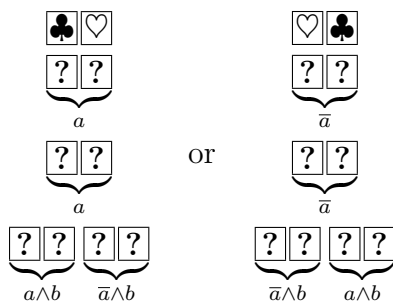
1. Turn all the cards face-down and rearrange the commitments as follows:

$$\underbrace{\boxed{?} \boxed{?}}_a \\
 \underbrace{\boxed{?} \boxed{?}}_0 \\
 \underbrace{\boxed{?} \boxed{?}}_0 \\
 \underbrace{\boxed{?} \boxed{?} \boxed{?} \boxed{?}}_0 \quad \underbrace{\hspace{1.5cm}}_b$$

2. Apply a pile-shifting shuffle as follows:

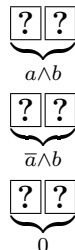


3. Reveal the top-most two cards to obtain two commitments to a and a commitment to $a \wedge b$ as follows:



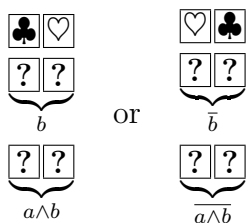
Note that the commitment next to that of $a \wedge b$ represents $\bar{a} \wedge b$ [32].

4. Remove the two commitments to a and rearrange the remaining cards as follows:



5. Apply a pile-shifting shuffle as in Step 2.

6. Reveal the top-most two cards to obtain commitments to b and $a \wedge b$ as follows:



Note that the commitment to $\bar{a} \wedge b$ becomes a commitment to $\bar{a} \wedge b \oplus a \wedge b = b$ (or \bar{b}) by shuffling in Step 5 and revealing the top-most two cards in Step 6 [32].

5.2. Setup phase

Both P and V place black commitments on all the cells of the grid. They also place “dummy” commitments (namely, red commitments) around the grid so that the grid is now of size $(p + 1) \times (q + 1)$.

5.3. Connectivity phase

P and V apply the protocol given in Sect. 2.4 with the removal of Step 2 of *Forming white-polyomino* from Sect. 2.4 by swapping to white each commitment of numbered cell. A white-polyomino is formed according to P 's solution. Now, V is convinced that the white commitments form a connected shape.

Now, V reveals the commitment corresponding to every numbered cell (which is public information). They must be white, otherwise V aborts. At this point, rule 3 is checked.

5.4. Verification phase

There are two rules to check; first V verifies that no black cells are touching (horizontally or vertically). For this, V considers each pair of adjacent cell and compute the *five-card trick* of Sect. 2.2 applied to OR. If any output is 1 (meaning that there are two black cells) then V aborts. Note that this verification is the same as the *Lonely Black* verification of [33].

The last rule to check is the rule 4, concerning the numbered cells, as follows.

1. P and V repeat the following steps for each numbered cell:
 - (a) Consider the cells on the four directions (north, east, south, and west) and apply our subprotocol proposed in Sect. 5.1 to each direction reading from the numbered cell to the edge, so that black commitments form as the right figure in Fig. 4.
 - (b) P and V take the $p + q - 1$ commitments and shuffle them. If the number of white commitments corresponds to the numbered cell then continue; otherwise abort.
2. If the previous step ends without aborting then V is convinced that rule 4 is fulfilled.

6. Security Proofs for Kurodoko

Our protocol for Kurodoko needs to verify three security properties given as theorems. Notice that we use the same security assumptions given in Sect.4

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

PROOF. First, notice that P convinces V in the sense that the protocol does not abort, which means that all the rules are satisfied. The protocol can be split in two: (1) the connectivity and (2) the verification phases.

(1) We use exactly the same technique as for Nurimisaki and [11] which proves the connectivity constraint.

(2) Because our subprotocol presented in Sect. 5.1 restores input commitments, we can consider the verification for each numbered cell as independent. So proving the completeness for one numbered cell is sufficient to prove the remaining ones. Since P knows the solution, the correct number of white commitments are placed around the numbered cell. The subprotocol allows to keep the white commitments close to the numbered cell and swap them if a black commitment split up them. Notice that we give all the cases in Fig. 5. This ensures that for each direction, the correct number of white commitments are kept regarding the numbered cell. \square

Theorem 5 (Soundness). *If P does not provide a solution of the $p \times q$ Kurodoko grid, P is not able to convince V .*

PROOF. Suppose that P does not know the solution; then, at least one of the rules is not verified. If the white cells are not connected then P cannot choose a black commitment at step 2; and hence, V will detect it. Notice that there is also the proof of this phase in [11].

The verification phase is now analysed; first rule 2: we apply the same technique as in [33] so the proof also holds in our case. Second, we consider rule 4 and show that the protocol will abort at the revealing phase. Here the correctness of the verification is directly given by the correctness of the subprotocol since one more white commitment will keep it white and one less will swap it to black. Since all the commitments for the four directions are revealed then the protocol will abort.

Notice that rule 3 is checked by revealing directly the commitments so the proof is trivial. \square

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

PROOF. We use the same proof technique as in [1], namely the description of an efficient *simulator* which simulates the interaction between an honest prover and a cheating verifier.

The simulator behaves as in [33] for the connectivity rule while we need to express it for the verification steps. The construction is simple since the last step for the numbered cell verification has a shuffle: the simulator only needs to put the correct number of white commitments (the simulator can simply keep the numbered cells white to fulfill rule 3). \square

7. Conclusion

We proposed physical ZKP protocols for Nurimisaki as well as for Kurodoko, that use only cards and envelopes.

The most difficult part was to prove that a cell is not a Misaki without leaking their color. Of course, we combined this part with the rest of the verifications that are stated by other rules. This new approach clearly demonstrates that showing that some cells do not have some properties is often more difficult than proving an explicit property without leaking any information.

Acknowledgements

We thank the anonymous referees, whose comments have helped us to improve the presentation of the paper. This work was supported in part by the Kayamori Foundation of Informational Science Advancement. This work was supported in part by JSPS KAKENHI Grant Numbers JP21K11881. This work was partially supported by the French ANR 18-CE39-0019 (MobiS5).

References

- [1] R. Gradwohl, M. Naor, B. Pinkas, G. N. Rothblum, Cryptographic and physical zero-knowledge proof systems for solutions of Sudoku puzzles, *Theory Comput. Syst.* 44 (2) (2009) 245–268. [doi:10.1007/s00224-008-9119-9](https://doi.org/10.1007/s00224-008-9119-9).
- [2] T. Sasaki, D. Miyahara, T. Mizuki, H. Sone, Efficient card-based zero-knowledge proof for Sudoku, *Theor. Comput. Sci.* 839 (2020) 135–142. [doi:10.1016/j.tcs.2020.05.036](https://doi.org/10.1016/j.tcs.2020.05.036).

- [3] S. Ruangwises, Two standard decks of playing cards are sufficient for a ZKP for Sudoku, *New Gener. Comput.* 40 (2022) 49–65. doi:[10.1007/s00354-021-00146-y](https://doi.org/10.1007/s00354-021-00146-y).
- [4] X. Bultel, J. Dreier, J. Dumas, P. Lafourcade, Physical zero-knowledge proofs for Akari, Takuzu, Kakuro and KenKen, in: E. D. Demaine, F. Grandoni (Eds.), *Fun with Algorithms*, Vol. 49 of *LIPICs*, 2016, pp. 8:1–8:20. doi:[10.4230/LIPICs.FUN.2016.8](https://doi.org/10.4230/LIPICs.FUN.2016.8).
- [5] D. Miyahara, T. Sasaki, T. Mizuki, H. Sone, Card-based physical zero-knowledge proof for Kakuro, *IEICE Trans. Fundamentals* 102-A (9) (2019) 1072–1078. doi:[10.1587/transfun.E102.A.1072](https://doi.org/10.1587/transfun.E102.A.1072).
- [6] X. Bultel, J. Dreier, J. Dumas, P. Lafourcade, D. Miyahara, T. Mizuki, A. Nagao, T. Sasaki, K. Shinagawa, H. Sone, Physical zero-knowledge proof for Makaro, in: T. Izumi, P. Kuznetsov (Eds.), *SSS 2018*, Vol. 11201 of *LNCS*, Springer, Cham, 2018, pp. 111–125. doi:[10.1007/978-3-030-03232-6_8](https://doi.org/10.1007/978-3-030-03232-6_8).
- [7] J.-G. Dumas, P. Lafourcade, D. Miyahara, T. Mizuki, T. Sasaki, H. Sone, Interactive physical zero-knowledge proof for Norinori, in: D.-Z. Du, Z. Duan, C. Tian (Eds.), *Computing and Combinatorics*, Vol. 11653 of *LNCS*, Springer, Cham, 2019, pp. 166–177. doi:[10.1007/978-3-030-26176-4_14](https://doi.org/10.1007/978-3-030-26176-4_14).
- [8] P. Lafourcade, D. Miyahara, T. Mizuki, L. Robert, T. Sasaki, H. Sone, How to construct physical zero-knowledge proofs for puzzles with a “single loop” condition, *Theor. Comput. Sci.* 888 (2021) 41–55. doi:[10.1016/j.tcs.2021.07.019](https://doi.org/10.1016/j.tcs.2021.07.019).
- [9] L. Robert, D. Miyahara, P. Lafourcade, T. Mizuki, Physical zero-knowledge proof for Suguru puzzle, in: S. Devismes, N. Mittal (Eds.), *Stabilization, Safety, and Security of Distributed Systems*, Vol. 12514 of *LNCS*, Springer, Cham, 2020, pp. 235–247. doi:[10.1007/978-3-030-64348-5_19](https://doi.org/10.1007/978-3-030-64348-5_19).
- [10] L. Robert, D. Miyahara, P. Lafourcade, L. Libralesso, T. Mizuki, Physical zero-knowledge proof and NP-completeness proof of Suguru puzzle, *Inf. Comput.* 285 (B) (2022) 104858. doi:[10.1016/j.ic.2021.104858](https://doi.org/10.1016/j.ic.2021.104858).
- [11] L. Robert, D. Miyahara, P. Lafourcade, T. Mizuki, Card-based ZKP for connectivity: Applications to Nurikabe, Hitori, and Heyawake, *New Gener. Comput.* 40 (2022) 149–171. doi:[10.1007/s00354-022-00155-5](https://doi.org/10.1007/s00354-022-00155-5).
- [12] S. Ruangwises, T. Itoh, Securely computing the n-variable equality function with 2n cards, *Theor. Comput. Sci.* 887 (2021) 99–110. doi:[10.1016/j.tcs.2021.07.007](https://doi.org/10.1016/j.tcs.2021.07.007).
- [13] S. Ruangwises, T. Itoh, Physical zero-knowledge proof for Numberlink puzzle and k vertex-disjoint paths problem, *New Gener. Comput.* 39 (1) (2021) 3–17. doi:[10.1007/s00354-020-00114-y](https://doi.org/10.1007/s00354-020-00114-y).
- [14] S. Ruangwises, T. Itoh, Physical ZKP for connected spanning subgraph: Applications to Bridges puzzle and other problems, in: I. Kostitsyna, P. Orponen (Eds.), *UCNC 2021*, Vol. 12984 of *LNCS*, Springer, Cham, 2021, pp. 149–163. doi:[10.1007/978-3-030-87993-8_10](https://doi.org/10.1007/978-3-030-87993-8_10).
- [15] R. Isuzugawa, D. Miyahara, T. Mizuki, Zero-knowledge proof protocol for Cryptarithmic using dihedral cards, in: I. Kostitsyna, P. Orponen (Eds.), *UCNC 2021*, Vol. 12984 of *LNCS*, Springer, Cham, 2021, pp. 51–67. doi:[10.1007/978-3-030-87993-8_4](https://doi.org/10.1007/978-3-030-87993-8_4).
- [16] S. Ruangwises, T. Itoh, How to physically verify a rectangle in a grid: A physical ZKP for Shikaku, in: P. Fraigniaud, Y. Uno (Eds.), *Fun with Algorithms*, Vol. 226 of *LIPICs*, Schloss Dagstuhl, 2022, pp. 24:1–24:12. doi:[10.4230/LIPICs.FUN.2022](https://doi.org/10.4230/LIPICs.FUN.2022).

- 24.
- [17] L. Robert, D. Miyahara, P. Lafourcade, T. Mizuki, Hide a liar: Card-based ZKP protocol for Usowan, in: D. Du, D. Du, C. Wu, D. Xu (Eds.), *Theory and Applications of Models of Computation*, Vol. 13571 of LNCS, Springer, Cham, 2022, pp. 201–217. [doi:10.1007/978-3-031-20350-3_17](https://doi.org/10.1007/978-3-031-20350-3_17).
 - [18] Y.-F. Chien, W.-K. Hon, Cryptographic and physical zero-knowledge proof: From Sudoku to Nonogram, in: P. Boldi, L. Gargano (Eds.), *Fun with Algorithms*, Vol. 6099 of LNCS, Springer, Berlin, Heidelberg, 2010, pp. 102–112. [doi:10.1007/978-3-642-13122-6_12](https://doi.org/10.1007/978-3-642-13122-6_12).
 - [19] S. Ruangwises, An improved physical ZKP for Nonogram, in: *COCOA*, Vol. 13135 of LNCS, Springer, Cham, 2021, pp. 262–272. [doi:10.1007/978-3-030-92681-6_22](https://doi.org/10.1007/978-3-030-92681-6_22).
 - [20] K. Saito, Physical zero-knowledge proof for the pencil puzzle Nurimisaki, Graduation Thesis, The University of Electro-Communications, Tokyo (2020).
 - [21] C. Iwamoto, T. Ide, Computational complexity of Nurimisaki and Sashigane, *IEICE Trans. Fundamentals* 103 (10) (2020) 1183–1192. [doi:10.1587/transfun.2019DMP0002](https://doi.org/10.1587/transfun.2019DMP0002).
 - [22] J. Kölker, Kurodoko is NP-complete, *Journal of Information Processing* 20 (3) (2012) 694–706. [doi:10.2197/ipsjjip.20.694](https://doi.org/10.2197/ipsjjip.20.694).
 - [23] S. Goldwasser, S. Micali, C. Rackoff, The knowledge complexity of interactive proof-systems, in: *STOC 1985*, ACM, New York, 1985, pp. 291–304. [doi:10.1145/22145.22178](https://doi.org/10.1145/22145.22178).
 - [24] L. Robert, D. Miyahara, P. Lafourcade, T. Mizuki, Card-based ZKP protocol for Nurimisaki, in: *SSS*, Vol. 13751 of LNCS, Springer, Cham, 2022, pp. 285–298. [doi:10.1007/978-3-031-21017-4_19](https://doi.org/10.1007/978-3-031-21017-4_19).
 - [25] K. Shinagawa, T. Mizuki, J. C. N. Schuldt, K. Nuida, N. Kanayama, T. Nishide, G. Hanaoka, E. Okamoto, Card-based protocols using regular polygon cards, *IEICE Trans. Fundamentals* 100-A (9) (2017) 1900–1909. [doi:10.1587/transfun.E100.A.1900](https://doi.org/10.1587/transfun.E100.A.1900).
 - [26] A. Nishimura, Y. Hayashi, T. Mizuki, H. Sone, Pile-shifting scramble for card-based protocols, *IEICE Trans. Fundamentals* 101-A (9) (2018) 1494–1502. [doi:10.1587/transfun.E101.A.1494](https://doi.org/10.1587/transfun.E101.A.1494).
 - [27] B. den Boer, More efficient match-making and satisfiability: The five card trick, in: J. Quisquater, J. Vandewalle (Eds.), *EUROCRYPT 1989*, Vol. 434 of LNCS, Springer, Berlin, Heidelberg, 1989, pp. 208–217. [doi:10.1007/3-540-46885-4_23](https://doi.org/10.1007/3-540-46885-4_23).
 - [28] D. Miyahara, L. Robert, P. Lafourcade, S. Takeshige, T. Mizuki, K. Shinagawa, A. Nagao, H. Sone, Card-based ZKP protocols for Takuzu and Juosan, in: M. Farach-Colton, G. Prencipe, R. Uehara (Eds.), *Fun with Algorithms*, Vol. 157 of LIPIcs, Schloss Dagstuhl, Dagstuhl, 2021, pp. 20:1–20:21. [doi:10.4230/LIPIcs.FUN.2021.20](https://doi.org/10.4230/LIPIcs.FUN.2021.20).
 - [29] T. Mizuki, H. Sone, Six-card secure AND and four-card secure XOR, in: X. Deng, J. E. Hopcroft, J. Xue (Eds.), *FAW 2009*, Vol. 5598 of LNCS, Springer, Berlin, Heidelberg, 2009, pp. 358–369. [doi:10.1007/978-3-642-02270-8_36](https://doi.org/10.1007/978-3-642-02270-8_36).
 - [30] A. Koch, S. Walzer, Foundations for actively secure card-based cryptography, in: M. Farach-Colton, G. Prencipe, R. Uehara (Eds.), *Fun with Algorithms*, Vol. 157 of

- LIPIcs, Schloss Dagstuhl, Dagstuhl, 2021, pp. 17:1–17:23. [doi:10.4230/LIPIcs.FUN.2021.17](https://doi.org/10.4230/LIPIcs.FUN.2021.17).
- [31] S. Ruangwises, T. Itoh, Physical zero-knowledge proof for Ripple Effect, *Theor. Comput. Sci.* 895 (2021) 115–123. [doi:10.1016/j.tcs.2021.09.034](https://doi.org/10.1016/j.tcs.2021.09.034).
- [32] T. Nishida, Y. Hayashi, T. Mizuki, H. Sone, Card-based protocols for any boolean function, in: R. Jain, S. Jain, F. Stephan (Eds.), *Theory and Applications of Models of Computation*, Vol. 9076 of LNCS, Springer, Cham, 2015, pp. 110–121. [doi:10.1007/978-3-319-17142-5_11](https://doi.org/10.1007/978-3-319-17142-5_11).
- [33] L. Robert, D. Miyahara, P. Lafourcade, T. Mizuki, Interactive physical ZKP for connectivity: Applications to Nurikabe and Hitori, in: L. D. Mol, A. Weiermann, F. Manea, D. Fernández-Duque (Eds.), *Connecting with Computability*, Vol. 12813 of LNCS, Springer, Cham, 2021, pp. 373–384. [doi:10.1007/978-3-030-80049-9_37](https://doi.org/10.1007/978-3-030-80049-9_37).

Appendix A. Number of copies

The number of calls to copy protocol can be expressed given the size of the grid $p \times q$. Indeed, we can split the cells in three categories: (1) cells at a corner, (2) cells at a border but not at a corner and (3) cells at the middle of the grid. First, notice that the copy protocol is called for the same number of neighbors the cell has. Thus, the copy protocol is run, given each type of cell:

corner: 2,
border: 3,
middle: 4.

Thus, by computing the total number of cells for each type, we can find the total number of calls to the copy protocol. The number of cell for each category, for a $p \times q$ grid, is:

corner: 4,
border: $2(p - 2) + 2(q - 2) = 2(p + q - 4)$,
middle: $(p - 2)(q - 2)$.

Finally, the total number of calls to the copy protocol N_c is:

$$\begin{aligned}
N_c &= 2 \times 4 + 3 \times 2(p + q - 4) + 4 \times (p - 2)(q - 2) \\
&= 8 + 6p + 6q - 24 + 4pq - 8p - 8q + 16 \\
&= 2(2pq - p - q).
\end{aligned}$$