Interactive Physical ZKP for Connectivity: Applications to Nurikabe and Hitori*

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

[1] LIMOS, University Clermont Auvergne, CNRS UMR 6158, France
[2] Graduate School of Information Sciences, Tohoku University, Sendai, Japan
[3] Cyberscience Center, Tohoku University, Sendai, Japan
[4] National Institute of Advanced Industrial Science and Technology, Tokyo, Japan

Abstract. During the last years, many Physical Zero-knowledge Proof (ZKP) protocols for Nikoli's puzzles have been designed. In this paper, we propose two ZKP protocols for the two Nikoli's puzzles called Nurikabe and Hitori. These two puzzles have some similarities, since in their rules at least one condition requires that some cells are connected to each other, horizontally or vertically. The novelty in this paper is to propose two techniques that allow us to prove such connectivity without leaking any information about a solution.

Keywords: Zero-knowledge proofs · Card-based secure two-party protocols · Puzzle · Nurikabe · Hitori.

1 Introduction

Zero-Knowledge Proofs (ZKP) were introduced by Goldwasser et al. [7]. Such a protocol has two parties: a prover P and a verifier V. The prover P wants to convince the verifier V that P knows the solution s of a problem without revealing any information about s. A ZKP must satisfy the following properties:
Completeness. If P knows 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, outputs of a simulator and outputs of the real protocol follow the same probability distribution.

In [5], the authors proved that for any NP-complete problem there exists an interactive ZKP protocol. A physical ZKP uses only physical algorithms with day-to-day objects such as cards, envelopes or bags while prohibiting large computations (i.e., no computer allowed). In 2007, the first physical ZKP was introduced for Sudoku [8], which is the most famous Nikoli's[‡] puzzle. In this paper we focus on two other Nikoli's puzzles, *Nurikabe* and *Hitori*.

In [10] solving even simple versions of Nurikabe was proven to be NP-complete. In [9] the authors proved that Hitori is also NP-complete. One might think that

* This paper appears in Proceedings of CiE 2021. The final authenticated version is available online at https://doi.org/10.1007/978-3-030-80049-9_37.
‡ Nikoli is a game publisher famously known for its Sudoku puzzle.

physical ZKP protocols for Nurikabe and Hitori 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 [6]; however, such a transformation is not practical because the overhead must be included in the transformation. Besides, the transformed ZKP protocol does not capture the property of a puzzle.

Contributions: In this paper, we present physical ZKP protocols for Nurikabe and Hitori using a deck of cards. Our protocols achieve no soundness error. That is, no malicious P who does not have a solution can convince V that it has a solution. Our work is inspired by [12], where P has to convince V of a single loop property. For Nurikabe and Hitori, we take a similar strategy to [12]. That is, P first increases the number of black (or white) cells one by one so that the resulting cells are guaranteed to satisfy the constraint of connectivity; then V verifies all the remaining constraints. We note that our protocols in this paper could not be constructed by simply adapting the existing technique [12].

We emphasize that our proposed protocols can be applied to a situation where Bob cannot solve by hand a Nurikabe or Hitori puzzle Alice created. In addition to such really practical applications, we believe that one can add our protocols (with others such as 3-Colorability one for instance) to introduce the notion of a ZKP system to non-experts such as high school students.

Related Work: Efficient physical ZKP protocols for Nikoli puzzles have been proposed: Sudoku [8, 19], Akari [2], Takuzu [2, 13], Kakuro [2, 14], Kenken [2], Makaro [3], Norinori [4], Slitherlink [12], Juosan [13], Suguru [16], Ripple Effect [18], and Numberlink [17]. An important step in this line of research is to achieve no soundness error.

Nurikabe's rule: This puzzle is formed by a rectangular grid where some cells contain numbers (Figure 1). The goal is to color some cells in black as follows:

1. Each numbered cell tells the number of continuous white cells surrounded by black cells. Such a region is called an *island*.
2. An island must contain only one numbered cell.
3. The black cells form a connected figure (called a *sea*).
4. The *sea* cannot form a 2×2 area.

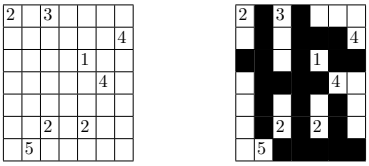


Fig. 1. Initial Nurikabe grid on the left and its solution on the right.

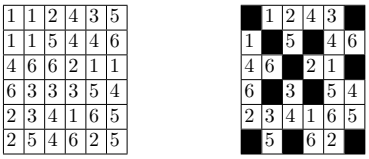


Fig. 2. Initial Hitori grid on the left and its solution on the right.

Hitori's rule: This puzzle is a grid where each cell contains a number as in the example of Figure 2. The goal is to color in black some cells with the following constraints:

1. Each row and each column must contain only one occurrence of a number.
2. The black cells cannot touch side to side although they can be diagonal.
3. The numbered cells must be connected to each other, horizontally or vertically.

2 Preliminaries

We introduce some notations of cards and shuffles and explain simple physical sub-protocols used in our constructions.

Card: A deck of cards used in our protocols consists of clubs $\boxed{\clubsuit}\boxed{\clubsuit}\cdots$, hearts $\boxed{\heartsuit}\boxed{\heartsuit}\cdots$, and number cards $\boxed{1}\boxed{2}\cdots$, whose backs are identical $\boxed{?}$. We encode three colors with the order of two cards as follows:

$$\text{black} \leftarrow \boxed{\clubsuit}\boxed{\heartsuit},\ \text{white} \leftarrow \boxed{\heartsuit}\boxed{\clubsuit},\ \text{red} \leftarrow \boxed{\heartsuit}\boxed{\heartsuit}. \tag{1}$$

We call such a face-down two 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*.

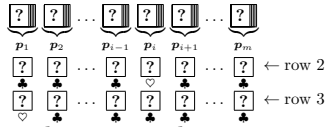
Pile-shifting shuffle [15,20]: This shuffling action means to *cyclically* shuffle piles of cards. More formally, given m piles, each of which consists of the same number of face-down cards, denoted by $(\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_m)$, applying a *pile-shifting shuffle* (denoted by $< \cdot | \cdots | \cdot >$) results in $(\boldsymbol{p}_{s+1}, \boldsymbol{p}_{s+2}, \ldots, \boldsymbol{p}_{s+m})$:

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

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 shuffle them by hand until nobody traces the offset.

Chosen pile protocol [4]: This is an extended version of the "chosen pile cut" proposed in [11]. Given m piles $(\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_m)$ with $2m$ additional cards, the *chosen pile protocol* enables a prover P to choose the i-th pile $\boldsymbol{p}_i$ and replace back the sequence of m piles to their original order.

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



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 the *row 2*. Then, one $\boxed{\heartsuit}$ appears, and the pile above the revealed $\boxed{\heartsuit}$ is the i-th pile (and hence, P can obtain $\boldsymbol{p}_i$). When this protocol is invoked, certain 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 the *row 2*. (Note, therefore, that we do not use the card $\boxed{\heartsuit}$ revealed in Step 3.) Then, apply a pile-shifting shuffle.
5. Reveal all the cards in the *row 3*. Then, one $\boxed{\heartsuit}$ appears, and the pile above the revealed $\boxed{\heartsuit}$ is $\boldsymbol{p}_1$. Therefore, by shifting the sequence of piles (such that $\boldsymbol{p}_1$ becomes the first 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 input sequence.

Input-preserving five-card trick [13]: Given two commitments to $a, b \in \{0, 1\}$ based on the encoding: $\boxed{\clubsuit}\boxed{\heartsuit} = 0$ and $\boxed{\heartsuit}\boxed{\clubsuit} = 1$, this sub-protocol [1, 13] starts by adding extra cards and rearranging the commitment to a so that we have the negation $\overline{a}$, as follows: $\underbrace{\boxed{?}\boxed{?}}_{a}\ \underbrace{\boxed{?}\boxed{?}}_{b} \rightarrow \underbrace{\boxed{?}\boxed{?}}_{\overline{a}}\ \boxed{\heartsuit}\ \underbrace{\boxed{?}\boxed{?}}_{b}\ \boxed{1}\boxed{2}\boxed{3}\boxed{4}\boxed{5}$.

The sub-protocol proceeds as follows to reveal only the value of $a \wedge b$ as well as restore commitments to a and b:

1. Rearrange the sequence of cards and then turn over the face-up cards as:
$$\boxed{?}\boxed{?}\boxed{\heartsuit}\boxed{?}\boxed{?}\boxed{1}\boxed{2}\boxed{3}\boxed{4}\boxed{5} \rightarrow \begin{matrix}\boxed{?}\boxed{?}\boxed{\heartsuit}\boxed{?}\boxed{?}\\ \boxed{1}\boxed{2}\boxed{3}\boxed{4}\boxed{5}\end{matrix} \rightarrow \begin{matrix}\boxed{?}\boxed{?}\boxed{?}\boxed{?}\boxed{?}\\ \boxed{?}\boxed{?}\boxed{?}\boxed{?}\boxed{?}\end{matrix}.$$
2. Regarding cards in the same column as a pile, apply a pile-shifting shuffle to the sequence: $\left\langle \begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix} \middle| \begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix} \middle| \begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix} \middle| \begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix} \middle| \begin{matrix}\boxed{?}\\ \boxed{?}\end{matrix} \right\rangle \rightarrow \begin{matrix}\boxed{?}\boxed{?}\boxed{?}\boxed{?}\boxed{?}\\ \boxed{?}\boxed{?}\boxed{?}\boxed{?}\boxed{?}\end{matrix}$.
3. Reveal all the cards in the first row, if the resulting sequence is:
   (a) $\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\heartsuit}\boxed{\heartsuit}\boxed{\heartsuit}$ (up to cyclic shifts), then we have $a \wedge b = 1$.
   (b) $\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\heartsuit}$ (up to cyclic shifts), then we have $a \wedge b = 0$.
4. After turning over all the face-up cards, apply a pile-shifting shuffle.
5. Reveal all the cards in the second row, i.e., all the number cards. Then, rearrange the sequence of piles so that the revealed number cards are in ascending order again to restore commitments to a and b.

3 ZKP Protocol for Nurikabe

We propose a ZKP protocol for Nurikabe, which is composed of three phases: the setup phase, the sea formation phase, and the verification phase. The full security proof is provided in (HAL), we only give here a sketch in Section 3.4.

Consider a puzzle instance of a $p \times q$ grid containing m numbered cells such that the ith numbered cell (in any order) has a number x_i for every i, $1 \leq i \leq m$. Remember that an island of a Nurikabe puzzle must contain exactly one numbered cell, and the number of white cells inside the island is indicated by the number written on the numbered cell. Thus, the number of (filled) black cells in the solution, denoted by N_b, is the difference between the number of total cells and the white cells (including the numbered cells), so $N_b = pq - \sum_{i=1}^{m} x_i$.

Thus, this number N_b can be regarded as public information, and indeed, we use the number N_b explicitly in our protocol.

Before going into the details of our protocol, let us define a *neighbour* cell and show a sub-protocol called the *4-neighbour protocol* that is important for constructing our ZKP protocols.

Neighbour cell: Consider a target cell c_t on a grid. A cell is a *neighbour* of c_t if it is next to c_t, on the left, the right, the top, or the bottom but not in diagonal.

4-neighbour protocol: Given pq commitments placed on a $p \times q$ grid, a prover P wants to reveal a target commitment and another one that lies next to the target commitment. Here, a verifier V is convinced that the second commitment is a neighbour of the first one (without knowing which one) as well as V confirms the colors of both the commitments. To handle the case where the target commitment is at the edge of the grid, we add red commitments (as "dummy" commitments) around the grid to prevent P from choosing a commitment that is not a neighbour. Thus, the size of the new grid is $(p+2) \times (q+2)$.

This protocol uses the chosen pile protocol (Section 2) twice. P first uses the chosen pile protocol to reveal a 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 the edge), the possible four neighbours are at distance 1 for the left or right one, and $p+2$ for the bottom or top one. Therefore, V and P can determine the positions of all the four neighbours. Among these, P chooses one commitment by using the chosen pile protocol again, and reveals it. This convinces V that the second commitment is indeed a neighbour. The rest of the protocol is to end the second and first chosen pile protocols.

3.1 Setup Phase

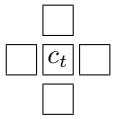
The verifier V and the prover P place a white commitment on each cell of a given $p \times q$ grid and place red commitments (as "dummy" commitments) around the grid so that we have $(p+2)(q+2)$ commitments on the board.

3.2 Sea Formation Phase

In this phase, P forms a sea on the board, i.e., P replaces a white commitment with a black commitment one by one according to the solution which only P knows, while hiding any information about the solution to V.

Let N_b be the number of black cells in the solution. This phase proceeds as follows.

1. P uses the chosen pile protocol to choose one white commitment which P wants to replace.
   (a) V reveals the chosen commitment; if it corresponds to white, V swaps the two cards constituting it so that the two cards become a black commitment. Otherwise, V aborts.
   (b) P and V end the chosen pile protocol to return the commitments to their original positions.
2. Repeat the following steps exactly $N_b - 1$ times:
   (a) P chooses one black commitment as a target and one white commitment among its neighbours using the 4-neighbour protocol; the neighbour is chosen such that P wants to make it black.



   (b) V reveals the target commitment. If it corresponds to black, V continues; otherwise V aborts.
   (c) V reveals the neighbour commitment (chosen by P). If it corresponds to white, V swaps the two cards constituting it to make it be a black commitment; otherwise V aborts.
   (d) P and V end the 4-neighbour protocol.
3. P and V replace every red commitment (i.e., dummy commitment) with a black commitment.

After this process, V is convinced that all the black commitments form a connected sea (rule 3).

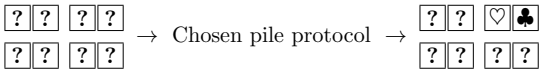
3.3 Verification Phase

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

Sea rule: Forbidden area. The prover P wants to convince V that any 2×2 area contains at least one white 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 squares.

Thus, P and V consider each 2×2 area of commitments one by one (in any order) and will repeat the following for each possible square:

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



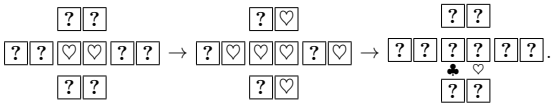
Island rules. P wants to convince V that the white cells respect the constraints. There are two verifications to make. Only one numbered cell for a given region and all white commitments are connected inside the region. Those two constraints are verified in the following protocol:

Let $n \geq 2$ be the number written on a given numbered cell.[§]

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

Now, V is convinced that the size of the island consisting of white cells is greater than or equal to n. To show that it is equal to n, it suffices to prove that there exists no white cell around them, as follows.

3. V replaces the commitment on the numbered cell with a black 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) Remember that P wants to show that any of four neighbour commitments is not white. Recall also the encoding (1), i.e., note that the right card of a black or red commitment is a heart $\boxed{\heartsuit}$. V reveals the right card of each of the four neighbours. If all of them are hearts (which means that all the commitments do not correspond to white), V replaces the chosen commitment with a black commitment; otherwise V aborts.



[§] For a numbered cell where 1 is written, V simply reveals the commitment on it and its four neighbours to confirm that the island is surrounded by the sea.

(d) P and V end the chosen pile protocol to return the commitments to their original positions.

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

3.4 Security Proofs

We give the following theorems to show that our protocol respects the security properties. All the proofs of our theorems are given in (HAL), we only give here a proof sketch.

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

Proof (sketch). We suppose that P knows the solution s of the grid and runs the setup phase. P is able to perform the proofs for the sea formation since all the black cells are connected. P is also able to end the verification phase. Basically, since s is a solution, all the rules are verified.

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

Proof (sketch). We suppose that P does not know the solution s and the proof is about showing that V will always detect it. Notice that the commitments of P form a connected figure (otherwise the protocol is ended without any verification). There are two cases to consider for the verification; (1) the forbidden area, if all the commitment are black on a 2×2 square then V will detect it since P cannot choose a white commitment, and (2) the island rules, where two invalid shapes can occur, when a region is completely covered with another region and, a part of a region is covered with another one. In both cases, we show that V will detect it using the protocol.

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

Proof (sketch). We use the same technique as in [8]; zero-knowledge is induced by a description of an efficient *simulator* which simulates interaction between a cheating verifier and a real prover. However, the simulator does not have a solution but it can swap cards for different ones during shuffles. The aim of the proof is to describe the behaviour of this simulator. Basically, the simulator creates a random connected figure of size N_b and during the verification, it swaps the cards to verify the rules.

4 ZKP Protocol for Hitori

We present a ZKP protocol for Hitori. The full security proof is provided in (HAL); we only give here a sketch presented in Section 4.4. Similar to our protocol for Nurikabe presented in Section 3, we let P choose a commitment which P wants to make white so that V is convinced that the resulting numbered cells are connected each other. However, we note that for Hitori the size of numbered cells could be information about the solution. That is, we cannot simply use the sea formation phase shown in Section 3.2. Therefore, we construct a sub-protocol called the *still-black protocol* as follows.

Still-black protocol: Given a black commitment, P can choose either changing it (i.e., swapping the two cards constituting the commitment) or not without V noticing it, as follows.

1. V reveals the given commitment to confirm that it is surely a black commitment.
2. If P wants to change the commitment, P places face-down club-to-heart below it; otherwise heart-to-club: $\boxed{?}\boxed{?} \rightarrow \begin{matrix}\boxed{?}\boxed{?}\\ \boxed{?}\boxed{?}\\ \clubsuit\ \heartsuit\end{matrix}$ or $\begin{matrix}\boxed{?}\boxed{?}\\ \boxed{?}\boxed{?}\\ \heartsuit\ \clubsuit\end{matrix}$.
3. Regarding cards in the same column as a pile, V applies a pile-shifting shuffle to the sequence of piles.
4. V reveals all the cards in the second row. If the revealed card on the right is a heart $\boxed{\heartsuit}$, V swaps the two cards in the first row; otherwise V does nothing.

4.1 Setup Phase

Put a black commitment on each cell of the $p \times q$ grid and red commitments around the grid.

4.2 Connectivity Phase

This phase follows the same steps as the ones in the sea formation phase shown in Section 3.2 (where a white commitment is regarded as a black one and vice versa) except for Step 2c; instead of swapping the two cards, V and P use the still-black protocol so that P can choose either swapping the two cards or not. (Remember that P cannot change a white commitment into black.) Note that the steps are repeated exactly $pq - 1$ times.

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

4.3 Verification phase

One occurrence for each row/column. Here, V checks if each row and column contains only one occurrence of a number. The idea is that for a given row or column it suffices to look at only numbered cells that appear $k > 1$ times and confirm that the k commitments on the numbered cells correspond to either k blacks or $k-1$ blacks. For a given row or column, this verification proceeds as follows.

1. V looks for numbered cells that appear more than once; take such a number which appears exactly $k > 1$ times. Then, V picks the corresponding k commitments.
2. P uses the chosen pile protocol to choose a white commitment among the k commitments if it exists; otherwise P uses the one to choose any commitment.
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. V and P end the chosen pile protocol to return the k commitments to their original places.
5. V and P repeat the above steps for all numbers that appear twice or more.

Lonely black. V checks that black cells are isolated from each other. Let a white commitment correspond to bit 0 and a black to 1. For each pair of adjacent commitments, V applies the input-preserving five-card trick (Section 2) to the two commitments. If the output is 0, V continues; otherwise V aborts.

4.4 Security Proofs

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

Proof (sketch). Suppose that P knows the solution; thus P can perform the connectivity and verification phases without aborting. There are two cases to consider, when P wants to change the black commitment and when P wants a black commitments to be still black.

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

Proof (sketch). Suppose that P does not know the solution. The proof consists in showing that V will detect it using the protocol. Without loss of generality, suppose that P gives correct commitments (i.e., white cells are connected) without corresponding to the solution, we show that V detects that the uniqueness and the lonely black constraints are not respected.

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

Proof (sketch). The same technique as for Nurikabe is used, namely the presence of a simulator that does not know the solution but can swap cards randomly.

5 Conclusion

We proposed two ZKP protocols for Nurikabe and Hitori. These two Nikoli's puzzles require that some cells of the solution are continuous without any precision on the number of cells in Hitori and without an exact number of cells in Nurikabe. We designed two methods and encoding for solving this continuity challenge and also respecting the other rules of the puzzles.

In the future, we aim at solving more challenging puzzles with other rules that also involve a kind of continuity property. For instance, in the puzzles Shikaku and Shakashaka, the goal is to draw rectangles of a certain size, which does not seem easy.

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 JSPS KAKENHI Grant Numbers JP19J21153 and JP21K11881. This study was partially supported by the French ANR project ANR-18-CE39-0019 (MobiS5). This work has been partially supported by the French government research program "Investissements d´Avenir" through the IDEX-ISITE initiative 16-IDEX-0001 (CAP 20-25) and the IMobS3 Laboratory of Excellence (ANR-10-LABX-16-01). This work was also supported by the French ANR project DECRYPT (ANR-18-CE39-0007) and SEVERITAS (ANR-20-CE39-0009).

References

1. den Boer, B.: More efficient match-making and satisfiability: *The Five Card Trick*. In: Quisquater, J., Vandewalle, J. (eds.) Advances in Cryptology – EUROCRYPT 1989. Lecture Notes in Computer Science, vol. 434, pp. 208–217. Springer, Berlin, Heidelberg (1989). https://doi.org/10.1007/3-540-46885-4_23
2. 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. 8:1–8:20. Schloss Dagstuhl, Dagstuhl (2016). https://doi.org/10.4230/LIPIcs.FUN.2016.8
3. 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. Lecture Notes in Computer Science, vol. 11201, pp. 111–125. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03232-6_8
4. 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. Lecture Notes in Computer Science, vol. 11653, pp. 166–177. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26176-4_14
5. 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
6. 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

7. Goldwasser, S., Micali, S., Rackoff, C.: Knowledge complexity of interactive proof-systems. Annual ACM Symposium on Theory of Computing pp. 291–304 (1985). https://doi.org/10.1145/3335741.3335750
8. 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
9. Hearn, R.A., Demaine, E.D.: Games, Puzzles, and Computation. A. K. Peters, Ltd., USA (2009)
10. 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
11. 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. 17:1–17:23. Schloss Dagstuhl, Dagstuhl (2021). https://doi.org/10.4230/LIPIcs.FUN.2021.17
12. Lafourcade, P., Miyahara, D., Mizuki, T., Sasaki, T., Sone, H.: A physical ZKP for Slitherlink: How to perform physical topology-preserving computation. In: Heng, S., López, J. (eds.) Information Security Practice and Experience. Lecture Notes in Computer Science, vol. 11879, pp. 135–151. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-34339-2_8
13. 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. 20:1–20:21. Schloss Dagstuhl, Dagstuhl (2021). https://doi.org/10.4230/LIPIcs.FUN.2021.20
14. Miyahara, D., Sasaki, T., Mizuki, T., Sone, H.: Card-based physical zero-knowledge proof for Kakuro. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. **102-A**(9), 1072–1078 (2019). https://doi.org/10.1587/transfun.E102.A.1072
15. Nishimura, A., Hayashi, Y., Mizuki, T., Sone, H.: Pile-shifting scramble for card-based protocols. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. **101-A**(9), 1494–1502 (2018). https://doi.org/10.1587/transfun.E101.A.1494
16. Robert, L., Miyahara, D., Lafourcade, P., Mizuki, T.: Physical zero-knowledge proof for Suguru puzzle. In: Devismes, S., Mittal, N. (eds.) Stabilization, Safety, and Security of Distributed Systems. Lecture Notes in Computer Science, vol. 12514, pp. 235–247. Springer, Cham (2020), https://doi.org/10.1007/978-3-030-64348-5_19
17. Ruangwises, S., Itoh, T.: Physical zero-knowledge proof for Numberlink puzzle and k vertex-disjoint paths problem. New Generation Computing **39**, 3–17 (2021), https://doi.org/10.1007/s00354-020-00114-y
18. Ruangwises, S., Itoh, T.: Physical zero-knowledge proof for Ripple Effect. In: Uehara, R., Hong, S.H., Nandy, S.C. (eds.) WALCOM: Algorithms and Computation. Lecture Notes in Computer Science, vol. 12635, pp. 296–307. Springer, Cham (2021), https://doi.org/10.1007/978-3-030-68211-8_24
19. 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
20. 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. Fundam. Electron. Commun. Comput. Sci. **100-A**(9), 1900–1909 (2017). https://doi.org/10.1587/transfun.E100.A.1900