

産学連携と数理・暗号分野連携による カードベース暗号の深化と新境地III 格子グラフ上のトポロジーに対する カードベースプロトコル

※資料公開にあたり、一部の図の
削除と、p.17に追記を行いました

電気通信大学 宮原大輝

本発表は次の文献に基づく: Daiki Miyahara, Pascal Lafourcade, and Maxime Puys, Card-Based ZKP Protocols for Connectivity-Based Puzzles: Extending to Tree Structures with Application to Nurimeizu, FUN 2026, LIPIcs, vol. 366, pp. 35:1–35:13, 2026

ゼロ知識証明 (ZKP)

- 証明者 P は, ある問題の答えを知っていることを検証者 V に納得させたい
- P は, 答えそのものは V に (1ビットも) 見せたくない



P

		1			5	6	7	
	2				4	8		
6	7							
3				5				
				4			1	8
					8	2		9
					2	4		
	9	2			7		8	3
	6		1					2



V

Sudoku Puzzle

- P は各マスに3枚の数字カードを置く
- V は置かれたカードが解を表すか検証する

		1			5	6	7	
	2				4	8		
6	7							
3				5				
				4			1	8
					8	2		9
					2	4		
	9	2			7		8	3
	6		1					2

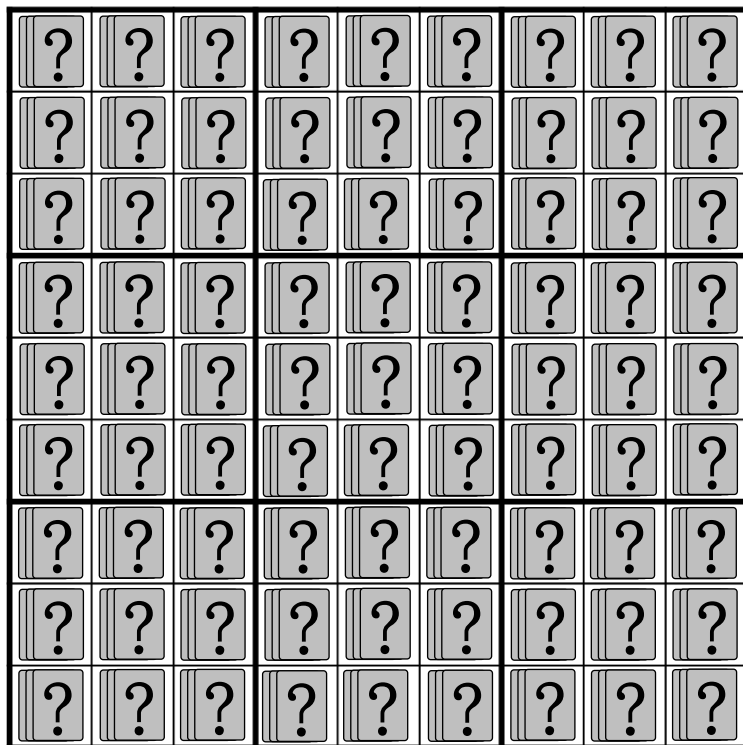
- P は各マスに3枚の数字カードを置く
- V は置かれたカードが解を表すか検証する

?	?	1	?	?	5	6	7	?
?	2	?	?	?	4	8	?	?
6	7	?	?	?	?	?	?	?
3	?	?	?	5	?	?	?	?
?	?	?	?	4	?	?	1	8
?	?	?	?	?	8	2	?	9
?	?	?	?	?	2	4	?	?
?	9	2	?	?	7	?	8	3
?	6	?	1	?	?	?	?	2

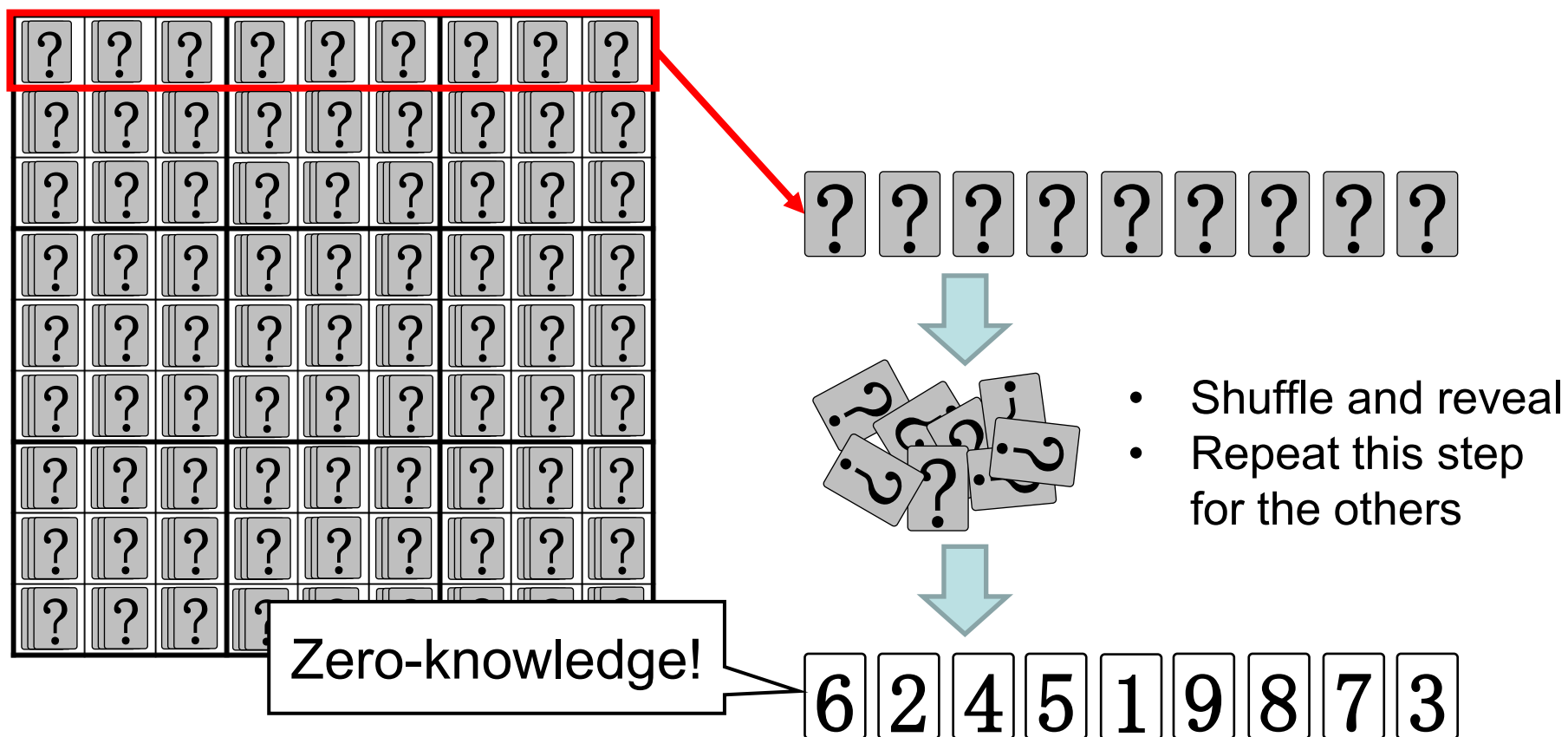
Face-down

Face-up

- P は各マスに3枚の数字カードを置く
- V は置かれたカードが解を表すか検証する

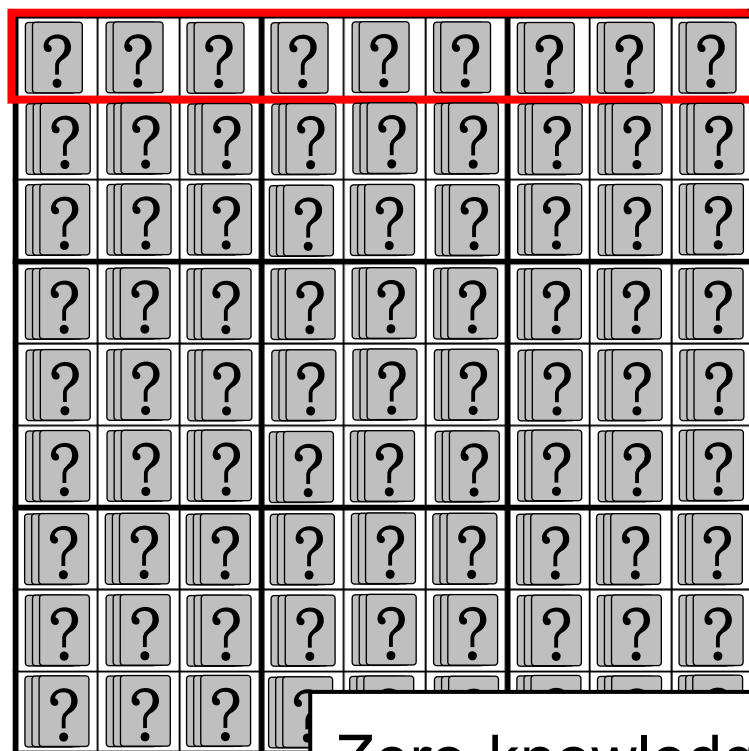


- P は各マスに3枚の数字カードを置く
- V は置かれたカードが解を表すか検証する



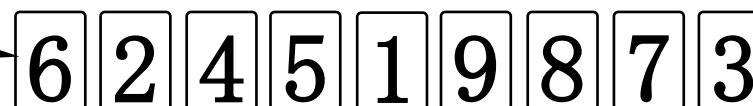
- P は各マスに3枚の数字カードを置く
- V は置かれたカードが解を表すか検証

Observation: V は3枚のカード
が同一か検証していない



- Shuffle and reveal
- Repeat this step for the others

Zero-knowledge!

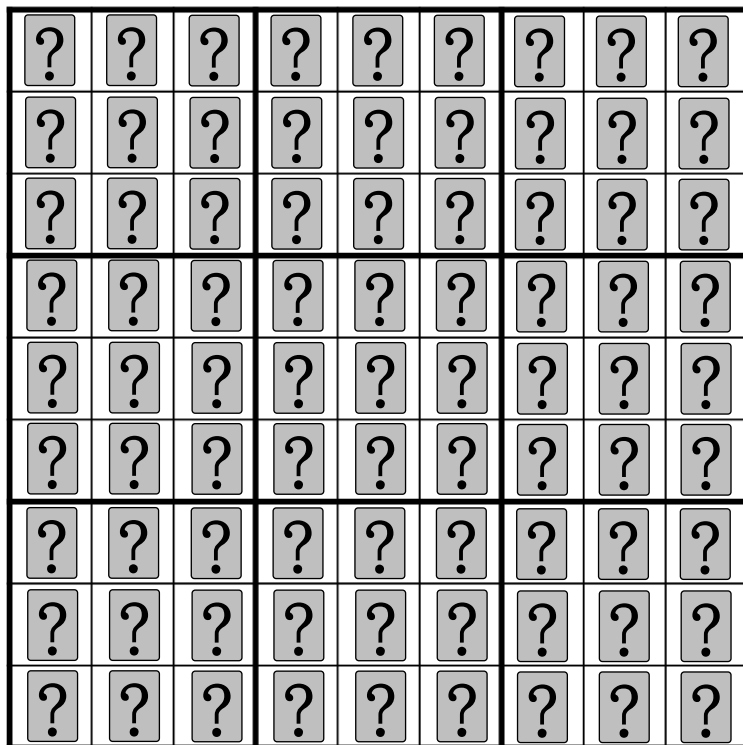


- P は各マスに1枚の数字カードを置く
- V は置かれたカードが解を表すか検証する

?	?	1	?	?	5	6	7	?
?	2	?	?	?	4	8	?	?
6	7	?	?	?	?	?	?	?
3	?	?	?	5	?	?	?	?
?	?	?	?	4	?	?	1	8
?	?	?	?	?	8	2	?	9
?	?	?	?	?	2	4	?	?
?	9	2	?	?	7	?	8	3
?	6	?	1	?	?	?	?	2

Only one card

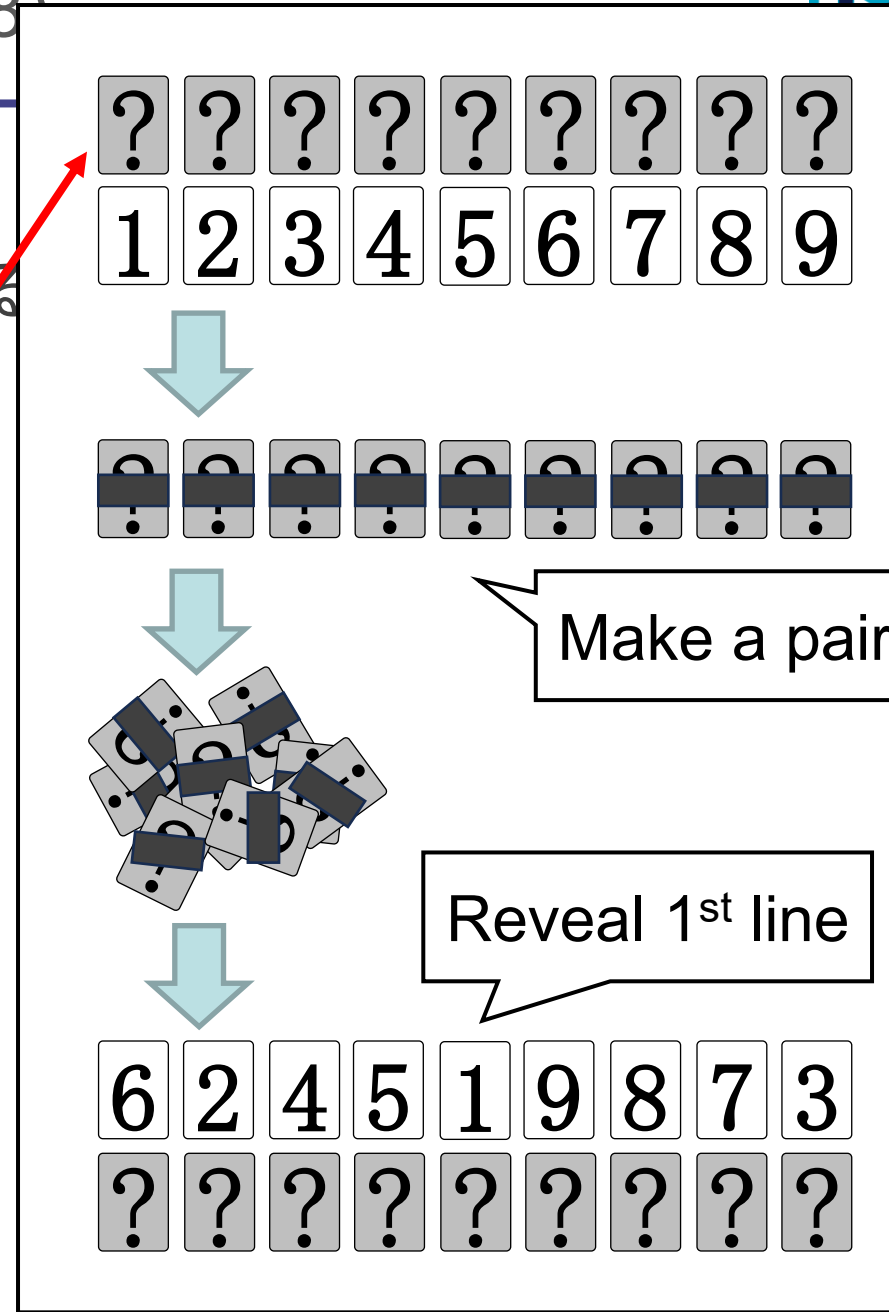
- P は各マスに1枚の数字カードを置く
- V は置かれたカードが解を表すか検証する



Protocol by Sasaki et al. (FUN 2018)

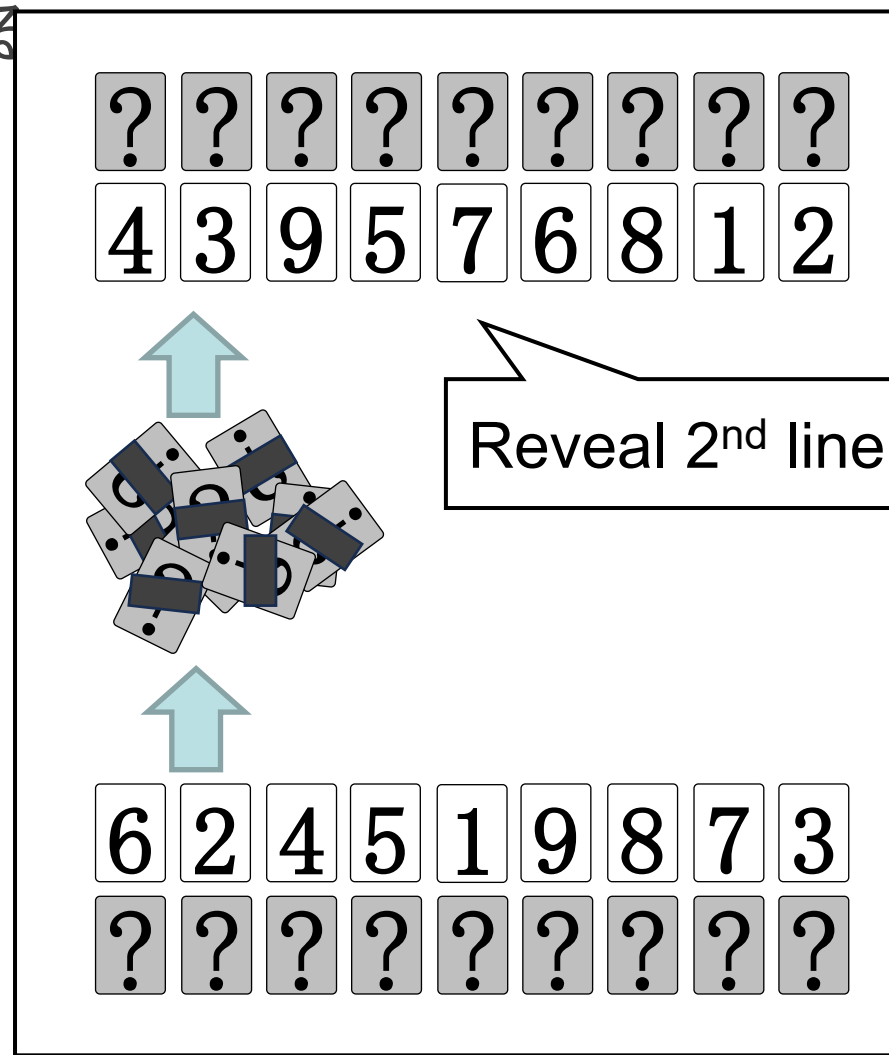
- P は各マスに1枚の数字カードを置く
- V は置かれたカードが解を表すか検証する

		1			5	6	7	
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?

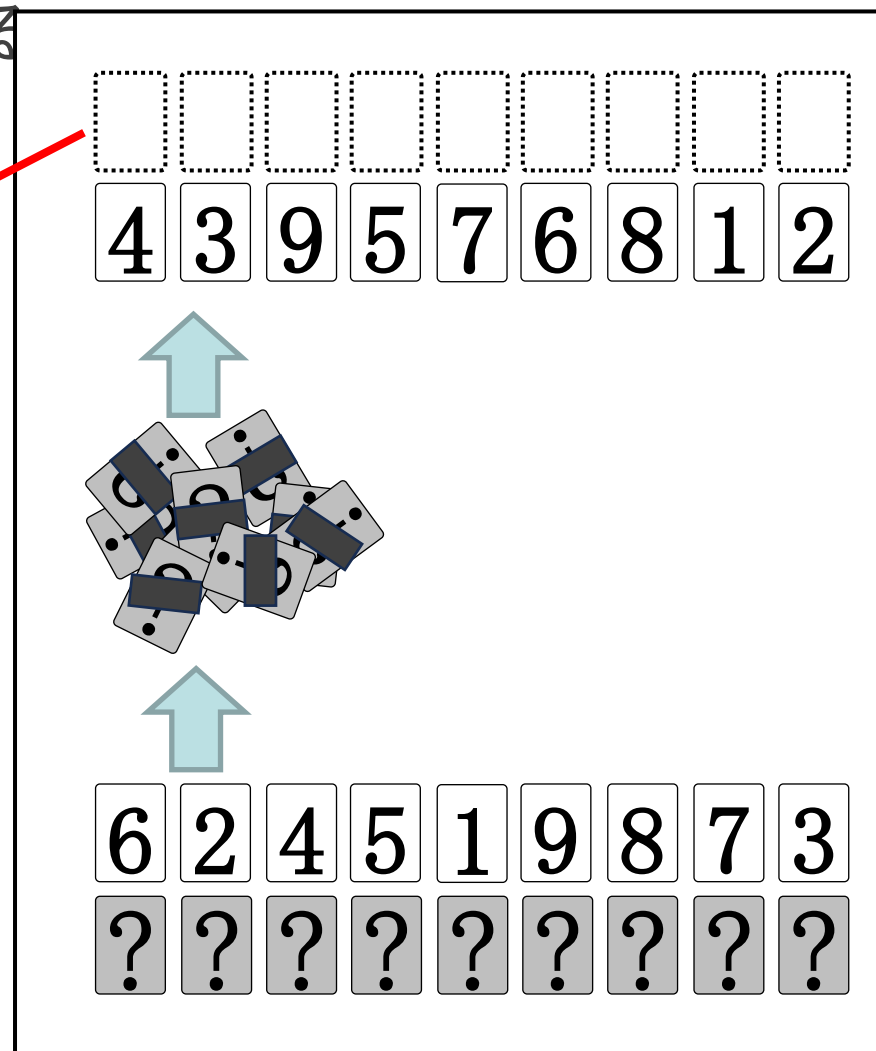
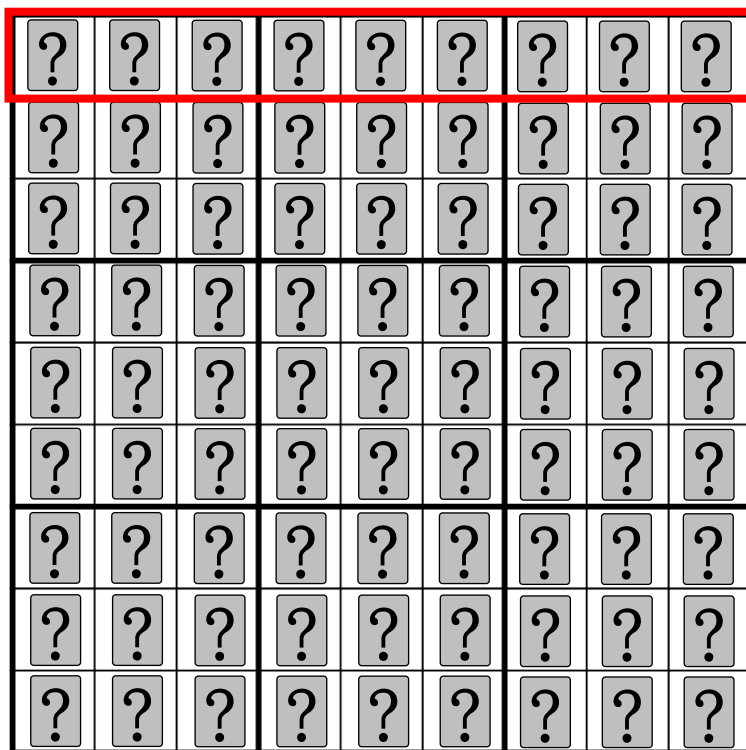


- P は各マスに1枚の数字カードを置く
- V は置かれたカードが解を表すか検証する

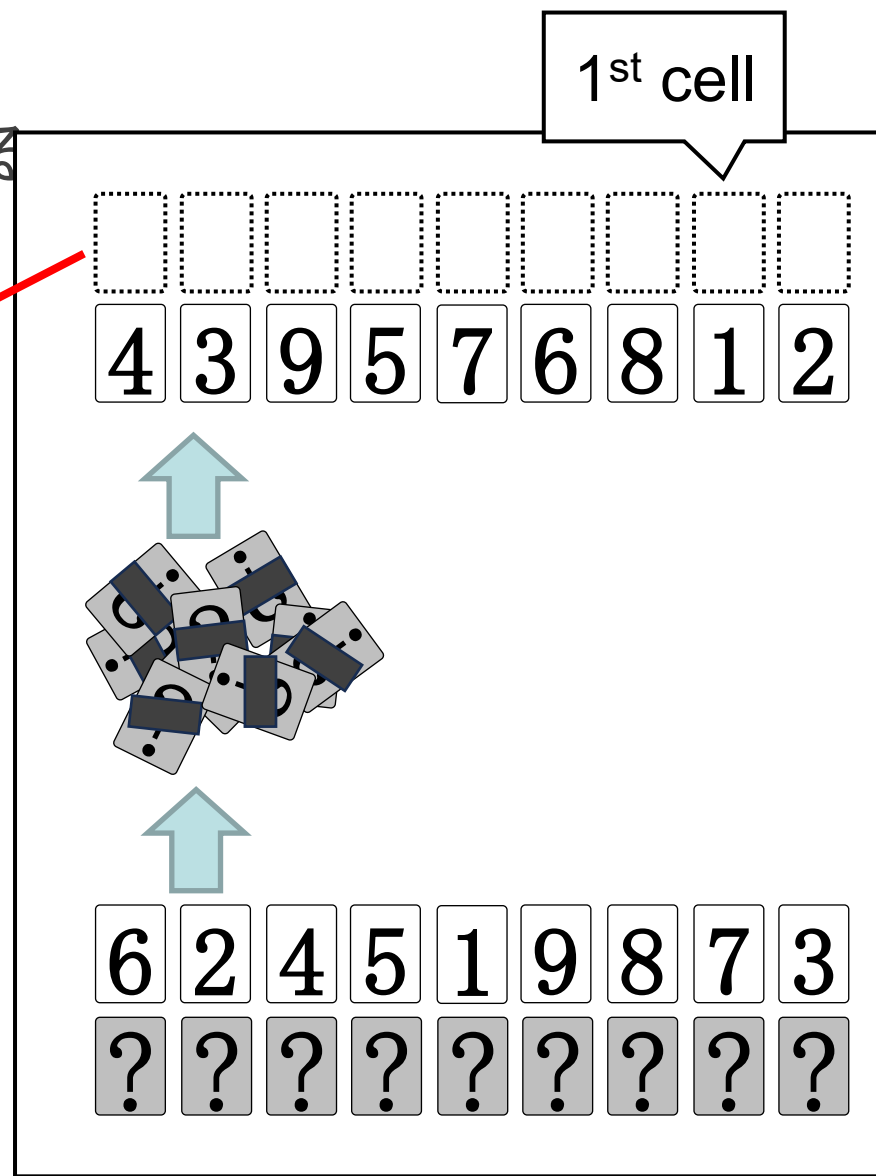
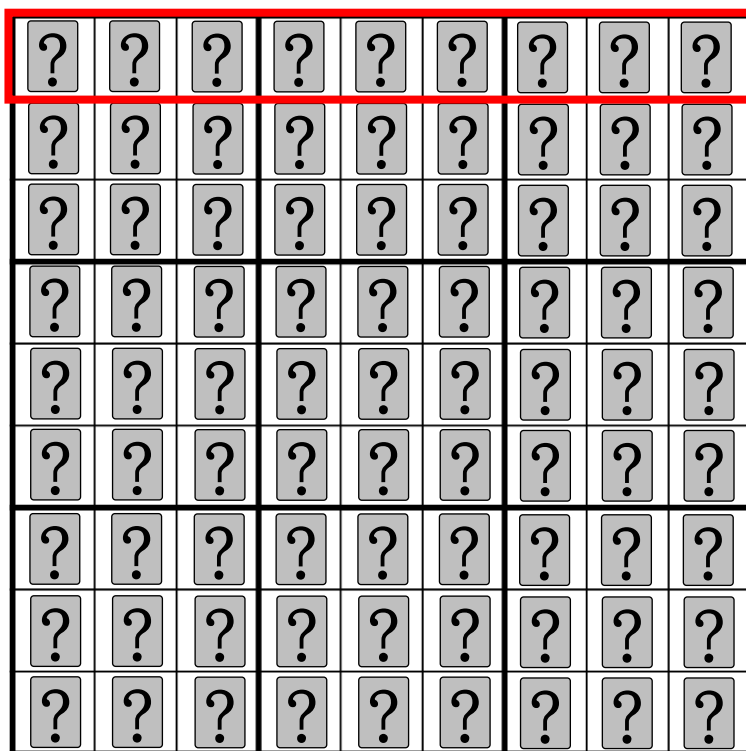
		1			5	6	7	
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?	?



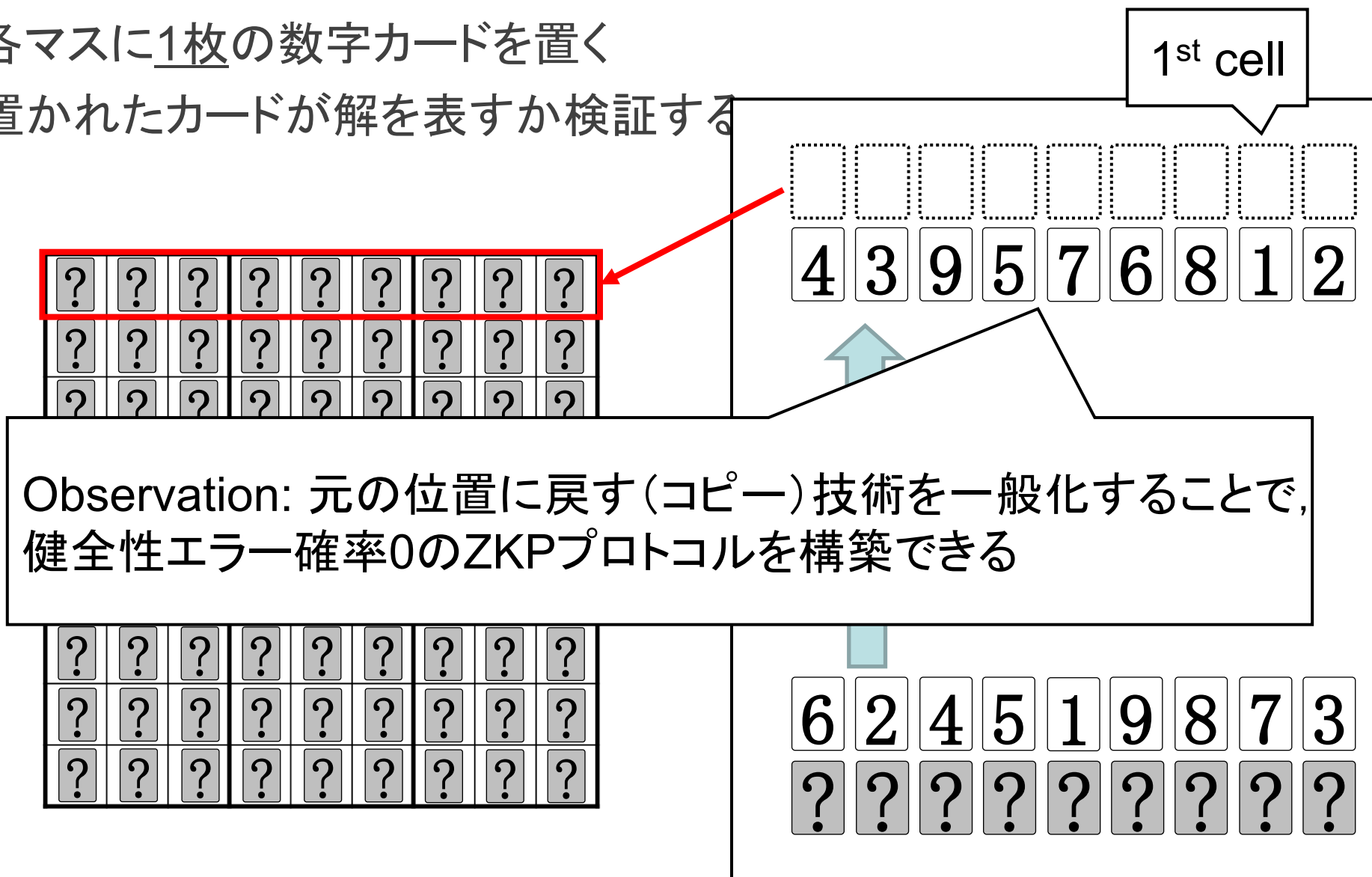
- P は各マスに1枚の数字カードを置く
- V は置かれたカードが解を表すか検証する



- P は各マスに1枚の数字カードを置く
- V は置かれたカードが解を表すか検証する



- P は各マスに1枚の数字カードを置く
- V は置かれたカードが解を表すか検証する



Puzzle	Authors in FUN Conferences
Sudoku	Gradwohl et al. (2007)
Nonogram	Chien and Hon (2010)
Akari, Takuzu, Kakuro, Kenken	Bultel et al. (2016)
Sudoku	Sasaki et al. (2018)
Takuzu, Juosan	Miyahara et al. (2020)
Shikaku	Ruangwises (2022)
Shakashaka	First talk of this session (2026)

- 様々なパズル・グラフ問題に対するZKPプロトコルが提案されてきた
- 幾何的な制約を持つパズル (ShikakuやShakashaka) が研究対象になっている

(株)ニコリが扱うパズルに対するZKPの最新状況

- ZKPプロトコルが構成されていない数字のパズルは16/55個

下線はZKPに関する発表がされていないパズル, 赤字はここ一年で取り組まれたパズル, 青字はここ一年で追加されたパズル, 文献は国際会議のみ

因子の部屋、ウソワン、お家へ帰ろう、カックロ、キンコンカン、クラリン、クロット、黒どこ(黒マスはどこだ)、基石ひろい、さしがね、さとがえり、サムライン、四角に切れ、シャカシャカ^[MRLK26]、シンカミノ、縦横さん、推理パズル、数コロ、数独、ストストーン、スラローム、スリザーリンク、ダブルチョコ、チェンブロ、チョコバナナ、月か太陽、ドッスンフワリ、ドッチループ、流れるループ、ナンスケ、ナンバーリンク、ぬりかべ、ぬりみさき、ぬりめいず^[MLP26]、のりのり、波及効果、橋をかけろ、バッグ、美術館、ひとりにしてくれ、フィルオミノ、ふくめん算、へびいちご、へやわけ、ヘルゴルフ、ペンシルズ、マカロ、ましゅ、マックロ、ミッドループ、虫くい算、やじさんかずさん、ヤジリン、よせなべ、LITS

MLP26 Daiki Miyahara, Pascal Lafourcade, and Maxime Puys, Card-Based ZKP Protocols for Connectivity-Based Puzzles: Extending to Tree Structures with Application to Nurimeizu, FUN 2026

MRLK26 Daiki Miyahara, Léo Robert, Pascal Lafourcade, and Shohei Kaneko, When Locality Implies Globality: Card-based ZKP Protocol for Shakashaka Puzzle, FUN 2026

(株)ニコリが扱うパズルに対するZKPの最新状況

- ZKPプロトコルが構成されていない言葉のパズルは20/21個

下線はZKPに関する発表がされていないパズル, 赤字はここ一年で取り組まれたパズル, 文献は国際会議のみ

アンドレ、おれひとりシークワーズ、書き取りスケルトン、カタカナ抜け文、カナオレ、漢字尻取り迷路、漢字部品マット、クロスワード、サムセレクション、シークワーズ、ジグソークロス、推理クロス、**スケルトンパズル**、**セレクトワーズ**、タめいロ、ドッキング漢字、ナンクロ、ネットワークス、ブロックパズル、余計文字クロス、よみどおり

- ZKPプロトコルが構成されていない絵のパズルは4/7個

浮き出しメイロ、同じの探し、**ステンドグラス**、**タイルペイント**^[GLMR25]、天体ショー、点つなぎ、まちがい探し

※発表時は抜けていました。他に雑誌で紹介されたパズルに対する構成も研究されています

- 連結領域, 木, 道, 閉路といったトポロジーを扱うサブプロトコルを提案
 - 格子グラフ G , パラメータ $\sigma \in \{\text{connected, tree, path, cycle}\}$ が与えられ, 頂点部分集合 $S \subseteq V(G)$ を構築
 - 誘導部分グラフ $G[S]$ が形状 σ を満たすことを保証, ただし S は(大きさ $|S|$ も含めて)秘匿
- 「ぬりめいず」の解に対するZKPプロトコルの構築に応用

Connected Region
[Robert et al. 2022]
[Sasaki and Shinagawa 2024]
[Nuida 2025]

Cycle
[Lafourcade et al. 2021]
[Hand et al. 2024]

Tree and Path
[This study]

※スリザーリンクは
辺を描くパズルの
ため, プロトコルを
変形する必要あり

文献	カード枚数	シャッフル数	グラフ	トポロジー	特徴
[Sasaki and Shinagawa 2024]	$O(k^2)$	$O(k^4)$	格子	連結	非対話
[Nuida 2025]	$O(k^2 \log k)^*$	$O(k^3 \log k)^*$	任意	連結(成分の数)	非対話
本研究	$O(k^2)$	$O(k^2)$	格子	連結, 木, 道, 閉路	対話

- [Sasaki and Shinagawa 2024]: 数コロに対するZKPを提案
 - 数コロ: 数字が入るマスは連結
 - P は最初にカードを置いた後, 何もしない(非対話型)
- [Nuida 2025]: 任意のグラフに対して, その連結成分を数えるプロトコルを提案
 - 連結成分の数=1の検証とすれば, 連結領域を描くパズルのZKPに応用可能
 - 非対話型
- 本研究: 格子グラフ上に連結領域, 木, 道, 閉路を(1つ)構築
 - P は V と対話しながら構築(対話的)
 - P が最初にカードを一気に置くことで非対話型に変換可能, ただし $O(k^4)$ 枚に増加

※サイズ $k \times k$ の格子
グラフに適用した場合

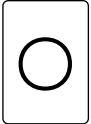
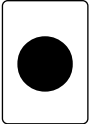
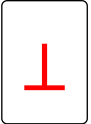
[Sasaki and Shinagawa 2024] Sasaki, S., Shinagawa, K. Physical Zero-Knowledge Proof for Sukoro. New Gener. Comput. 42, 381–398 (2024).

[Nuida 2025] Nuida, K. Card-Based Protocol Counting Connected Components of Graphs. New Gener. Comput. 43, 18 (2025).

- [Igari, Odaka, Komano, and Mizuki 2026]: ハミルトン閉路問題に対するZKP
 - 非対話型: P は部分グラフの辺リストを入力, V は辺リストの頂点を仮名化してシャッフル
 - シャッフル回数は3回
- [Fujiwara and Shinagawa 2026]: 上記の改善
 - 非対話型: P は部分グラフの隣接行列を(エンコードを工夫して)入力, V は行と列をシャッフル
 - 2回に削減
- 本研究: (格子グラフ上の)任意の頂点を通る閉路を構築
 - 対話型: 頂点数を秘匿
 - シャッフル回数は盤面サイズのオーダー

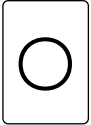
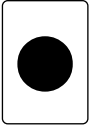
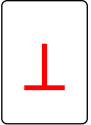
1. Introduction
2. Preliminaries
3. Proposed Protocol
4. Application to Nurimeizu
5. Conclusion

- 4種類のカードを用いる:

白:  , 黒:  , 番兵:  , 端: 

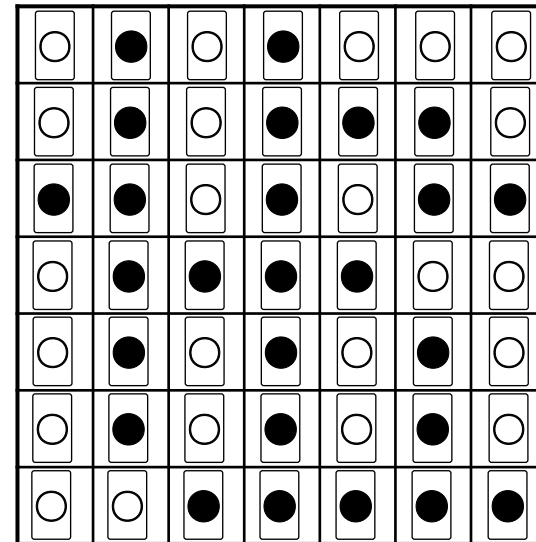
裏面はすべて同一とする: 

- 4種類のカードを用いる:

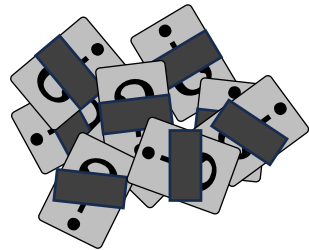
白: , 黒: , 番兵: , 端: 

裏面はすべて同一とする: 

- 白黒カードを用いることで, 解を表すことができる

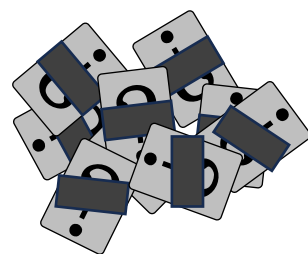


- Sasakiらのプロトコルで用いられたシャッフル操作:



パイルスクランブルシャッフル
と呼ばれる[Ishikawa et al. 2015]

- Sasakiらのプロトコルで用いられたシャッフル操作:



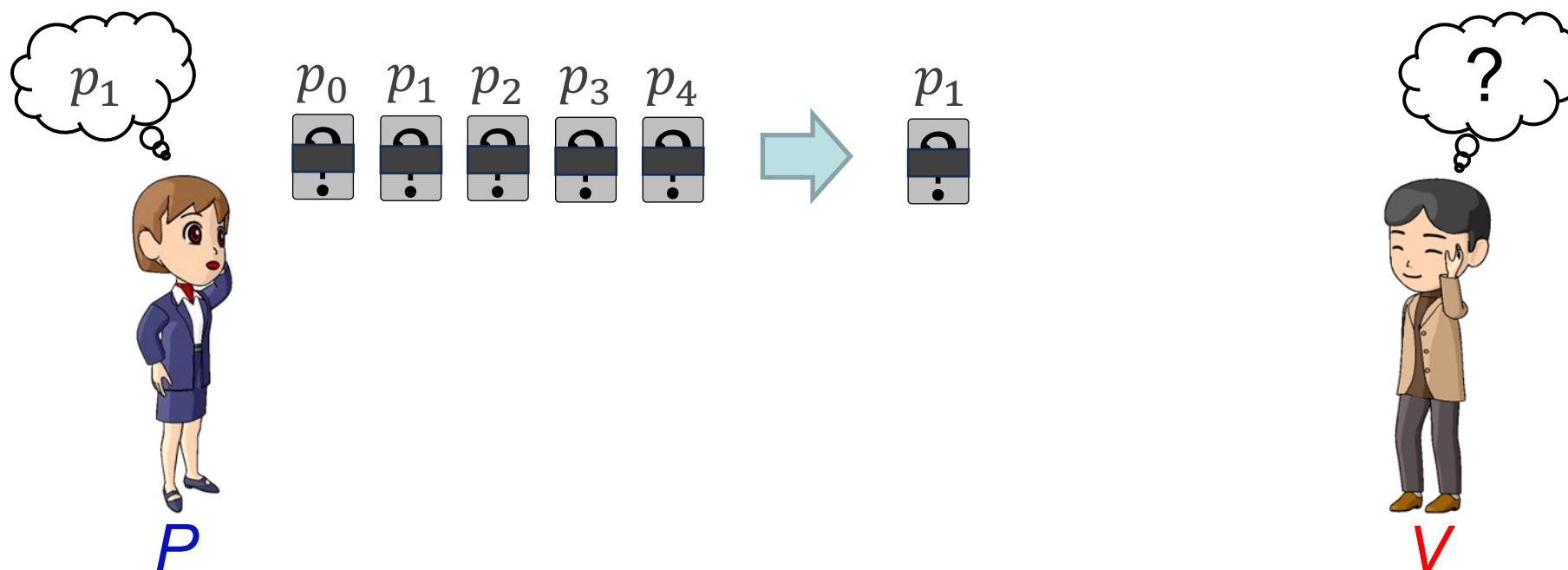
パイルスクランブルシャッフル
と呼ばれる[Ishikawa et al. 2015]

- 提案プロトコルではパイルシフティングシャッフルを用いる[Shinagawa et al, 2017]

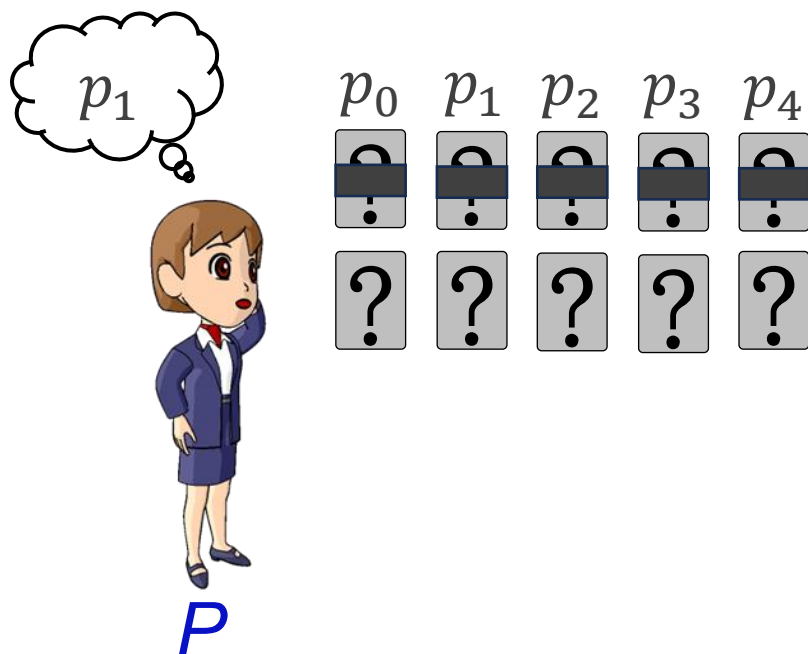


これはカード束を巡回的にランダムな回数だけシフトさせるシャッフル操作

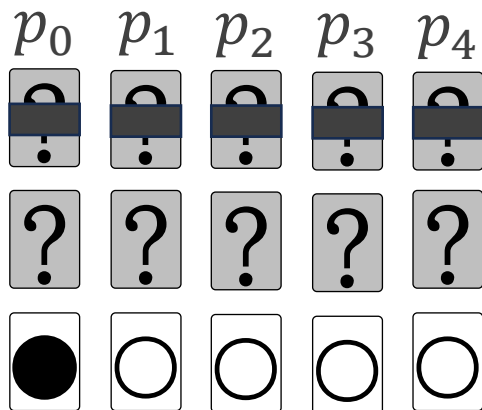
- k 個のカード束の列 $(p_0, p_1, \dots, p_{k-1})$ に対して, ある特定のカード束 p_i を P が選択
- ただし, そのインデックス i は V に秘匿される



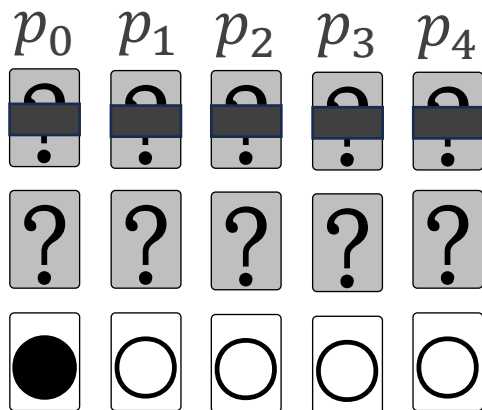
1. P は p_i の下に黒カード, それ以外のカードの下に白カードを裏にして置く(2行目)



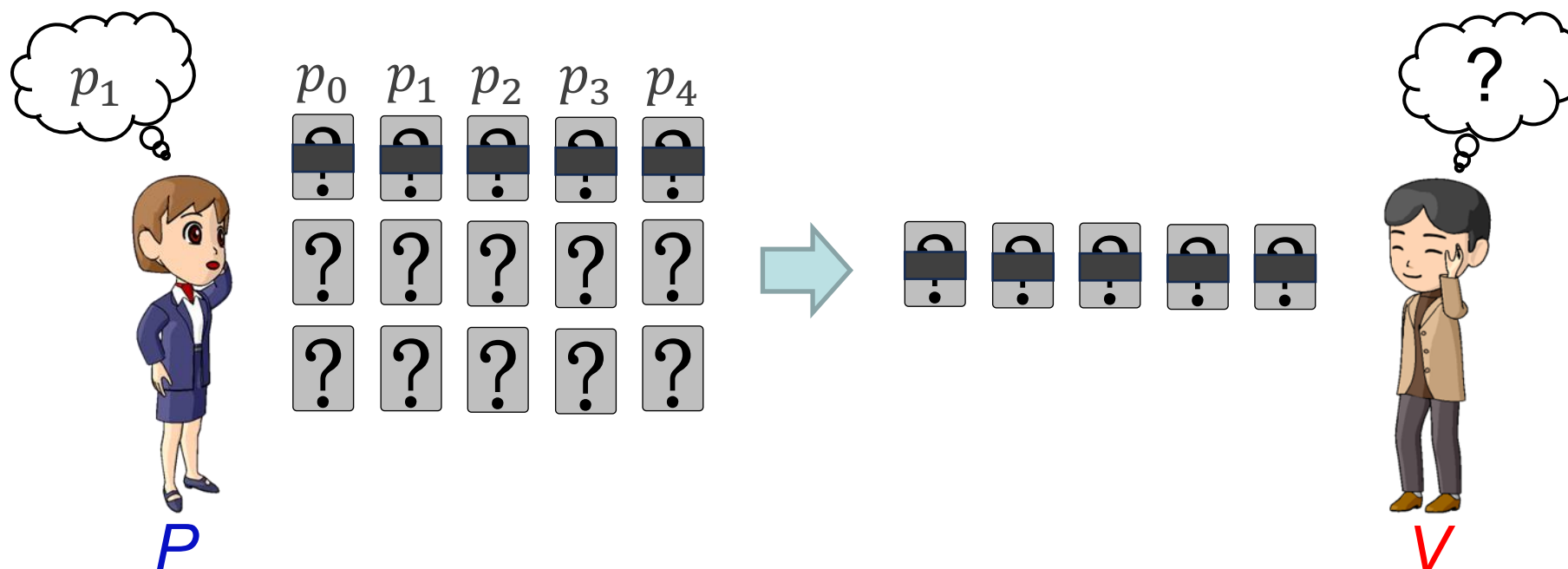
1. P は p_i の下に黒カード, それ以外のカードの下に白カードを裏にして置く(2行目)
2. p_0 の下に黒カード, それ以外のカードの下に白カードを表にして置く(3行目)



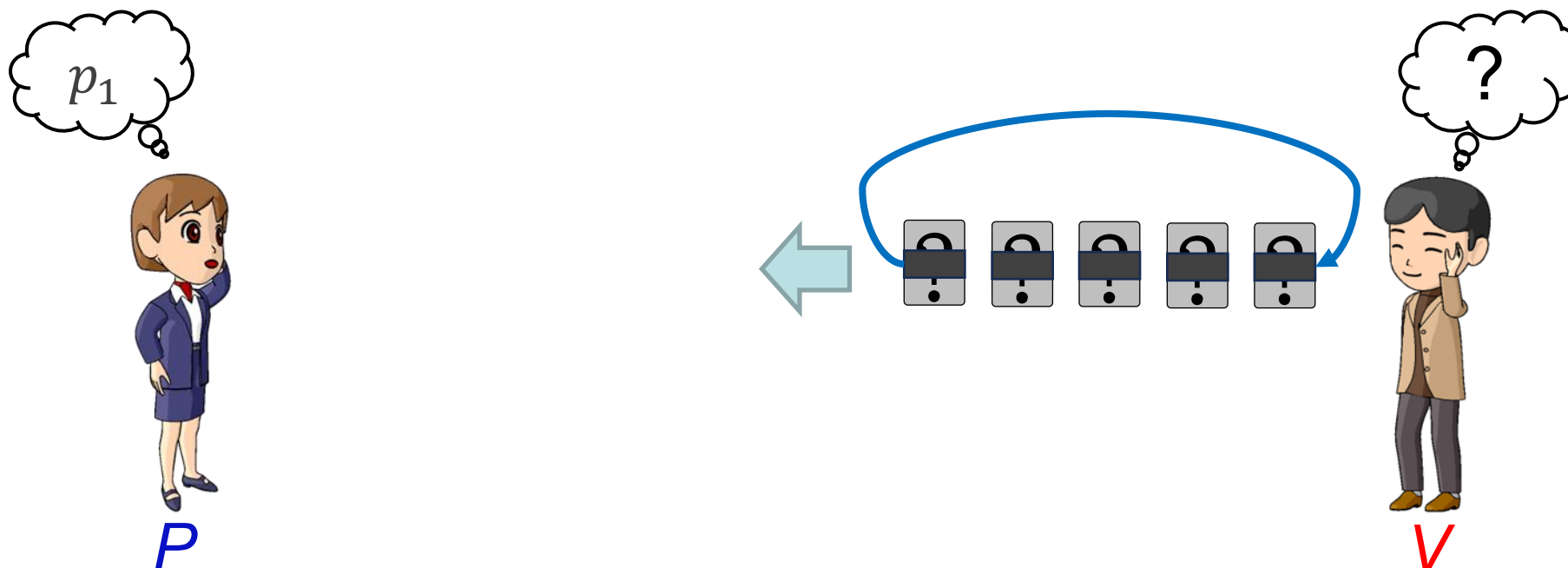
3. すべてのカードを裏にして、同じ列にあるカードを1つのカード束にする



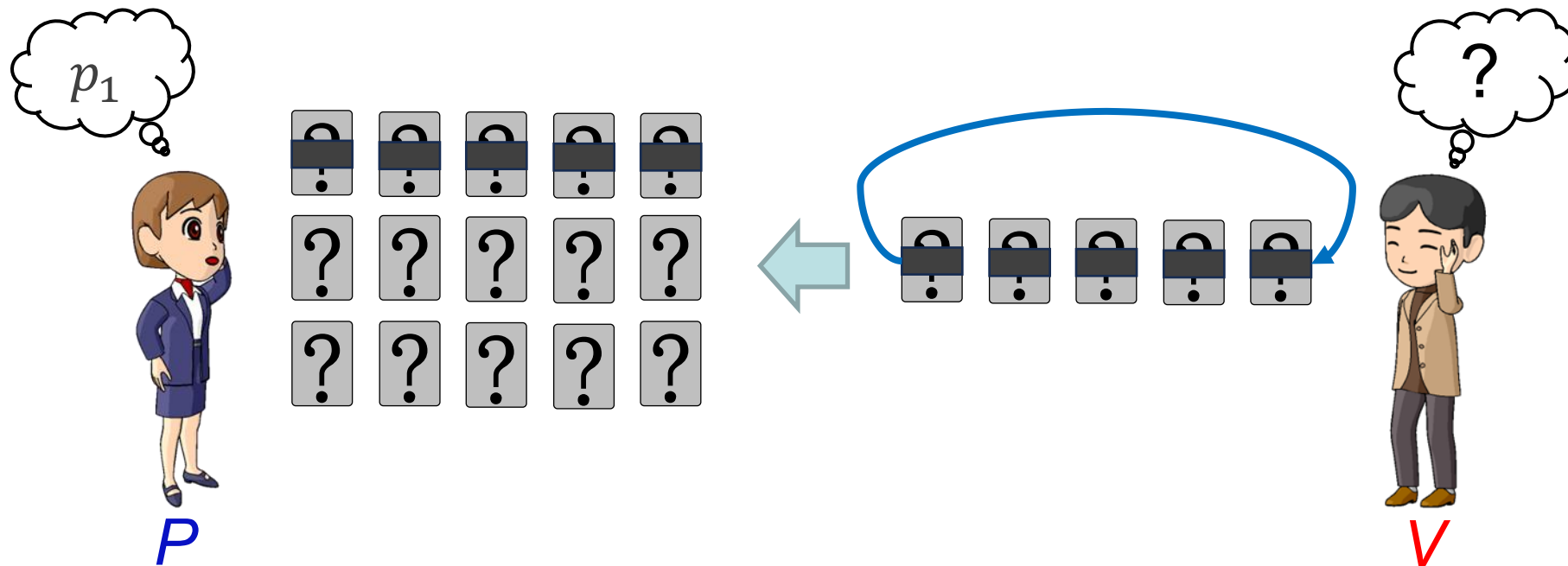
3. すべてのカードを裏にして、同じ列にあるカードを1つのカード束にする



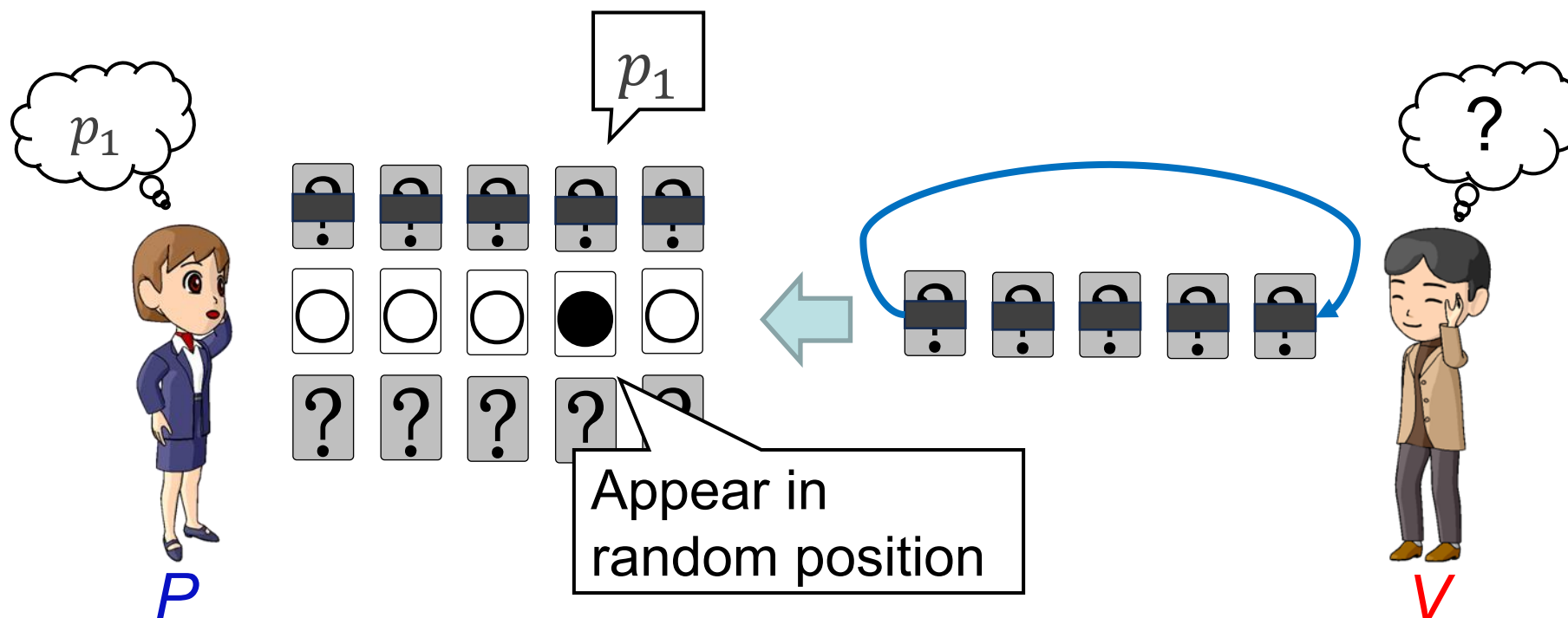
4. パイルシフティングシャッフルを行い、元の位置に並べる



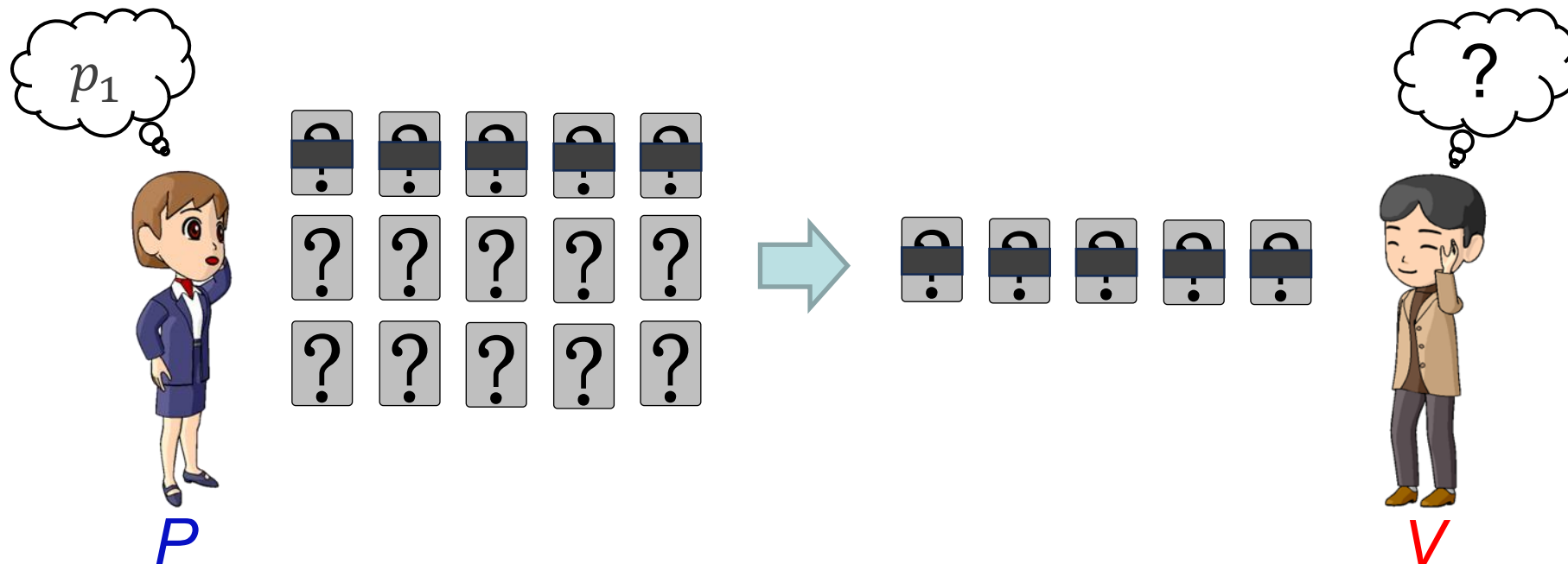
4. パイルシフティングシャッフルを行い、元の位置に並べる



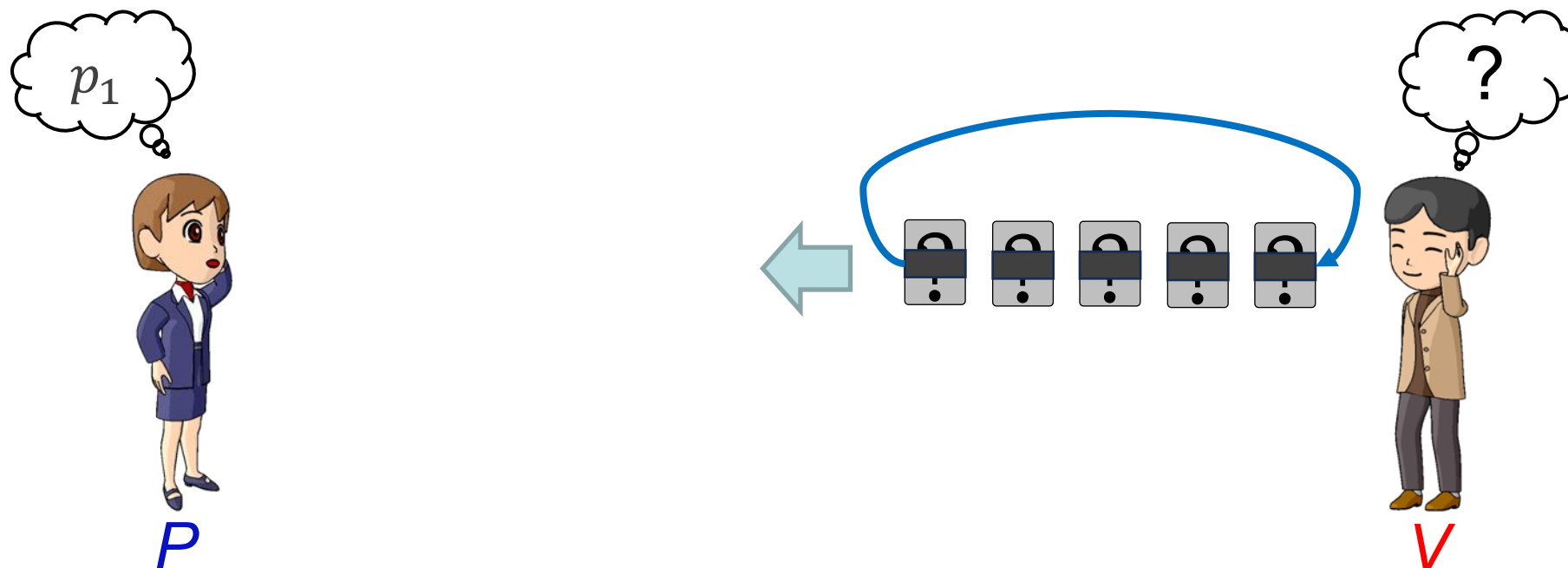
4. パイルシフティングシャッフルを行い、元の位置に並べる
5. 2行目のカードをすべてめくり、 p_i を出力する



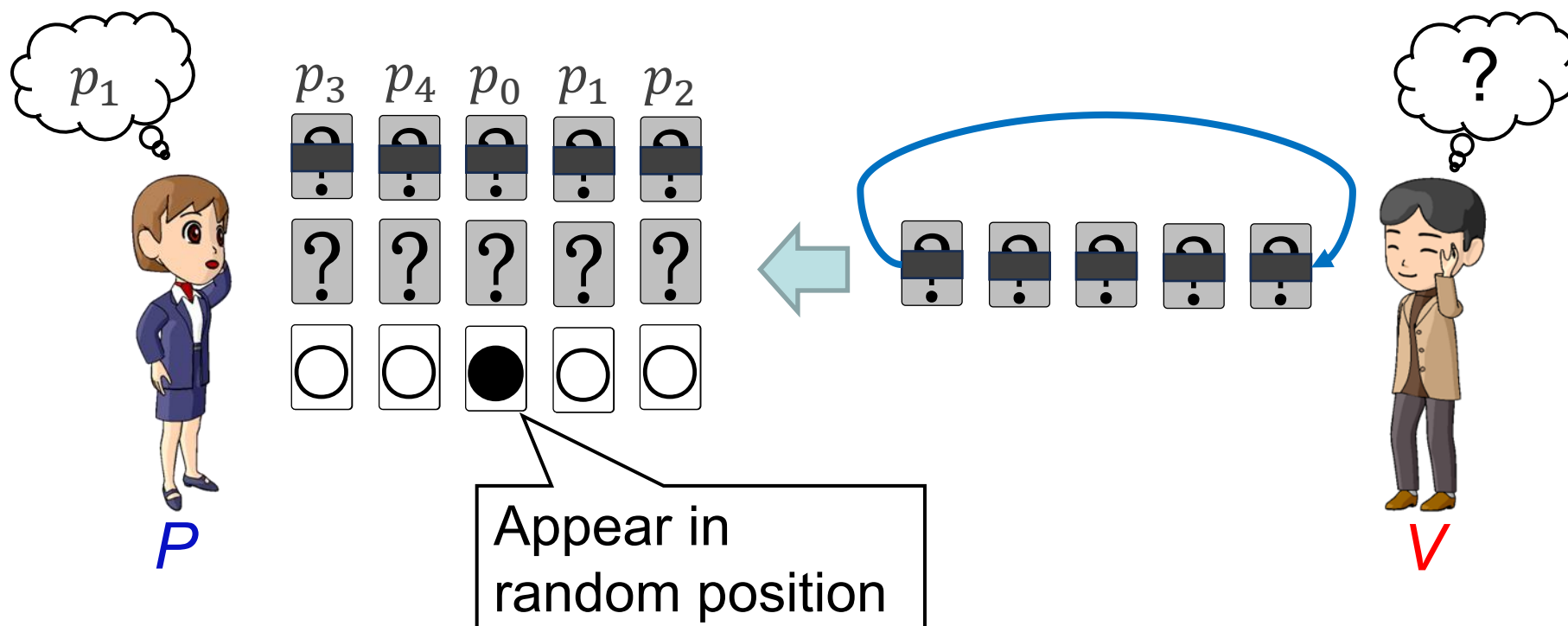
6. パイルシフティングシャッフルを再度行い、3行目のカードをすべてめくる



6. パイルシフティングシャッフルを再度行い、3行目のカードをすべてめくる



6. パイルシフティングシャッフルを再度行い、3行目のカードをすべてめくる



- 選択プロトコルをグリッド(格子グラフ)に適用
 - 各セルを一次元として巡回的に捉える
 - パイルシフティング後も, 各カード束 p_j の隣接関係が維持される
- グリッドの境界を示す番兵を追加する必要あり(後述)



p_0	p_1	p_2	p_3	p_4	p_5	p_6
p_7	p_8	p_9	p_{10}	p_{11}	p_{12}	p_{13}
p_{14}	p_{15}	...				
					...	p_{48}



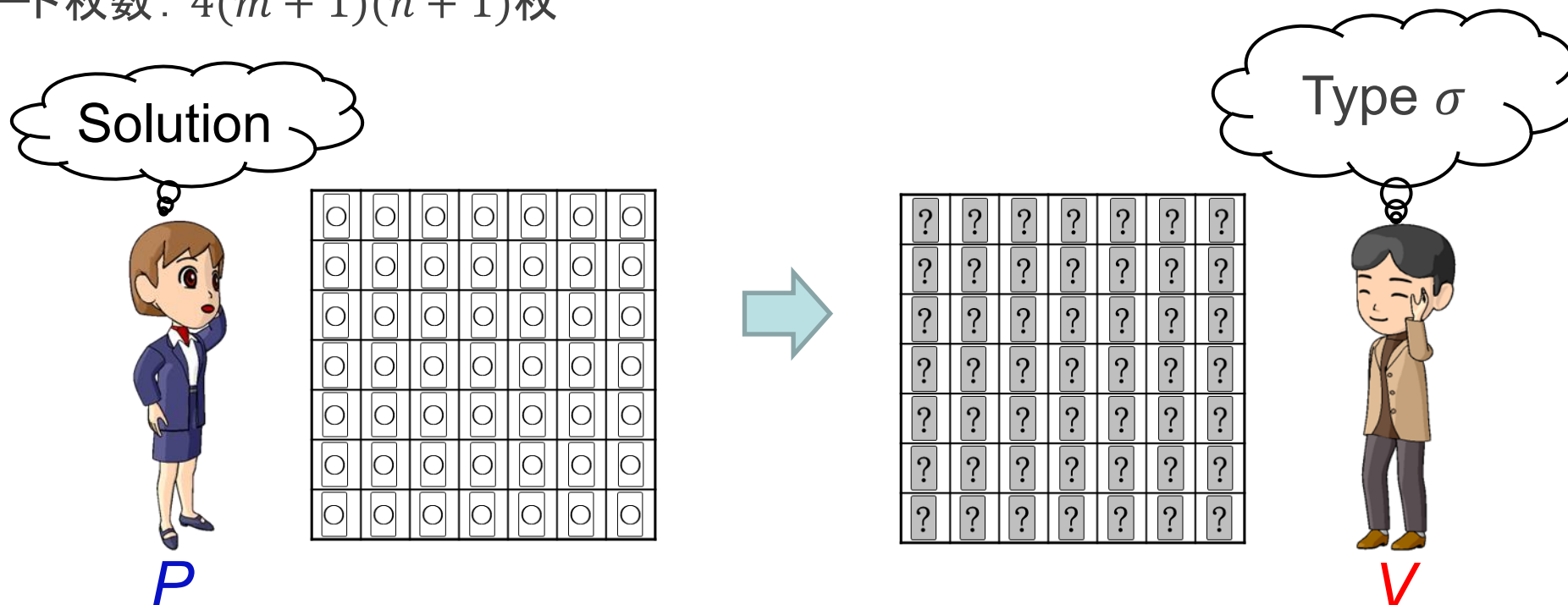
選択プロトコル

	...	p_{48}	p_0	p_1	p_2	p_3
p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
p_{11}	p_{12}	p_{13}	p_{14}	p_{15}	...	

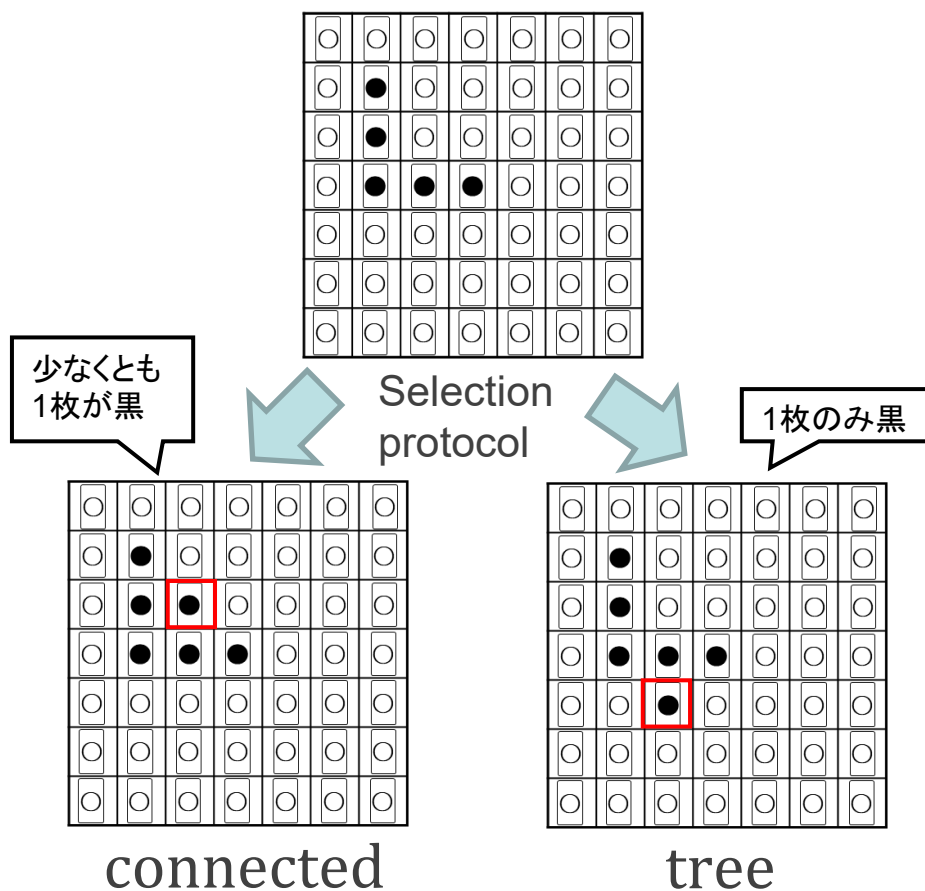


1. Introduction
2. Preliminaries
3. Proposed Protocol
4. Application to Nurimeizu
5. Conclusion

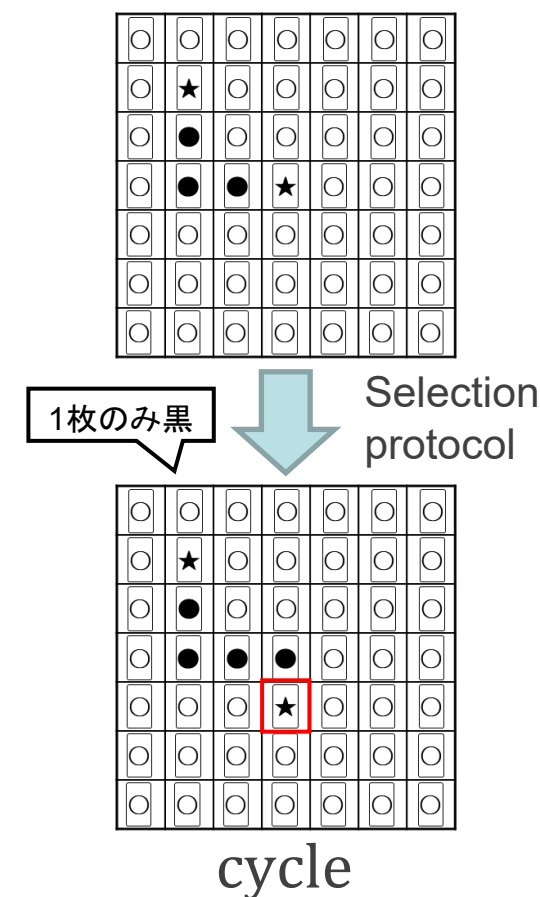
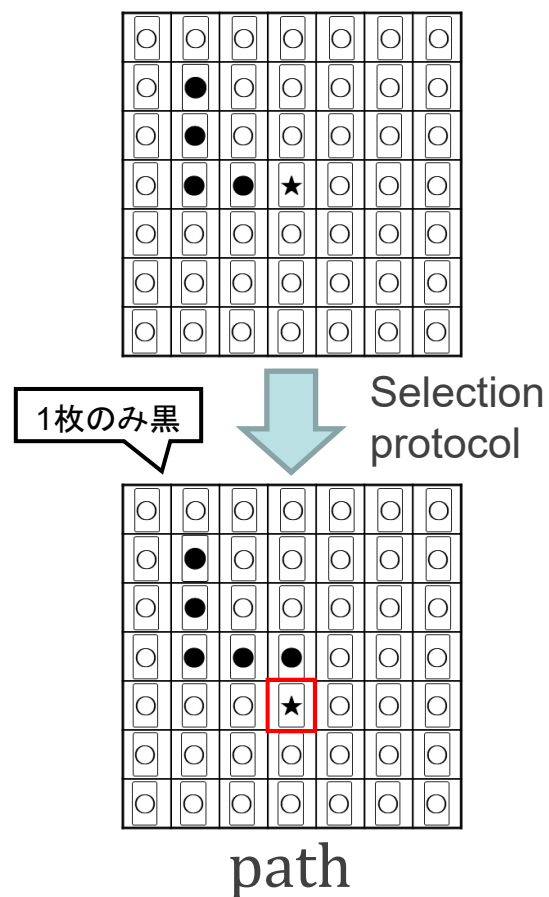
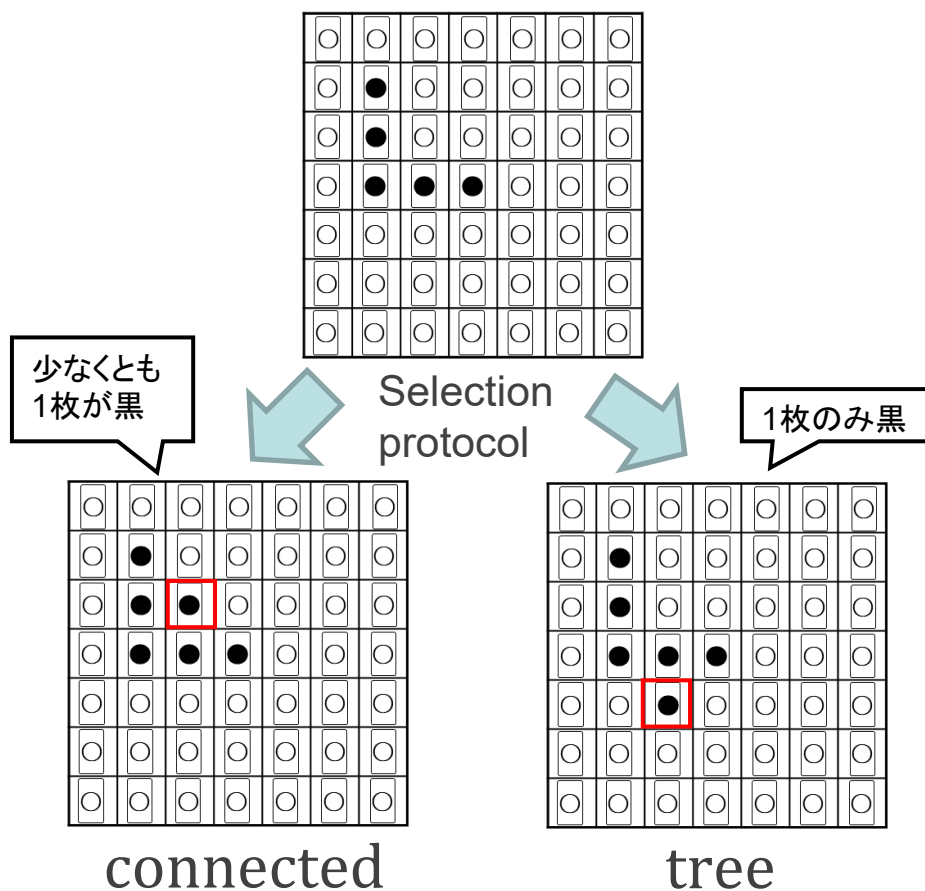
- 提案プロトコルを用いると, P は形状が $\sigma \in \{\text{connected}, \text{tree}, \text{path}, \text{cycle}\}$ であるように黒カードを配置できる. ただし具体的な配置は V に秘匿される
- 盤面サイズ $m \times n$ に対して, 選択プロトコルを $O(mn)$ 回使用する
 - $\sigma = \text{cycle}$: $3mn - 3$ 回, それ以外: $3mn - 4$ 回
 - カード枚数: $4(m + 1)(n + 1)$ 枚



- P は選択プロトコルを用いて白カードを選択し、黒・端カードに置き換える
- 隣接する4枚を検証し、その置き換えが形状 σ に従うことを保証する

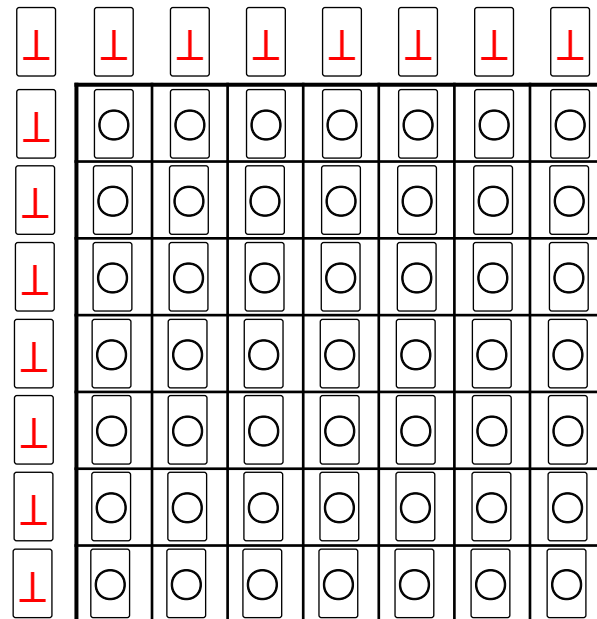


- P は選択プロトコルを用いて白カードを選択し、黒・端カードに置き換える
- 隣接する4枚を検証し、その置き換えが形状 σ に従うことを保証する



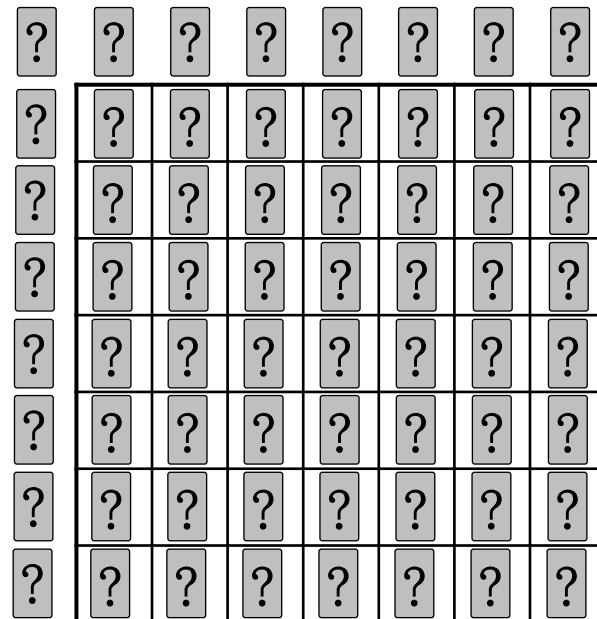
提案プロトコル(形状 σ)

1. 白カードを各セルに置き, 番兵カードを盤面の左側に置く



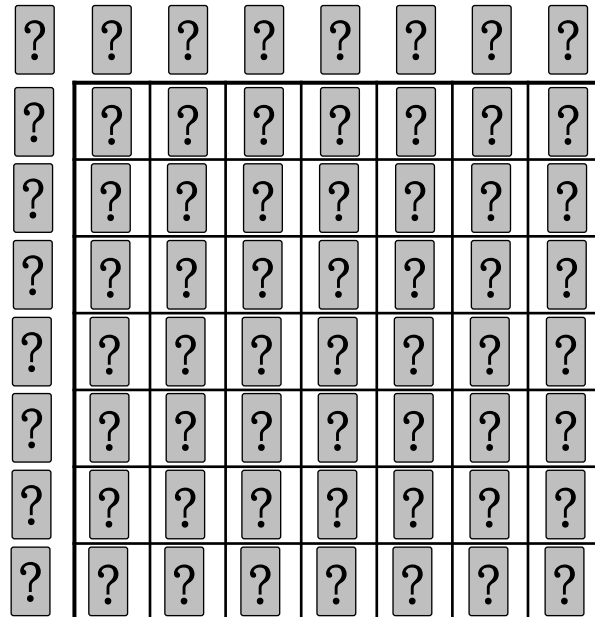
提案プロトコル(形状 σ)

1. 白カードを各セルに置き, 番兵カードを盤面の左側と上側に置く



提案プロトコル(形状 σ)

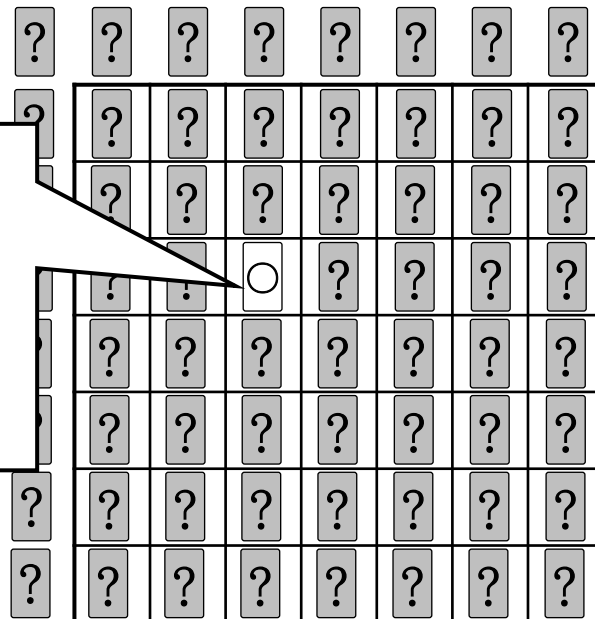
1. 白カードを各セルに置き, 番兵カードを盤面の σ 上側と左側に置く
2. 選択プロトコルを用いて, 白カードを選択する



提案プロトコル(形状 σ)

1. 白カードを各セルに置き, 番兵カードを盤面の上方と左側に置く
2. 選択プロトコルを用いて, 白カードを選択する

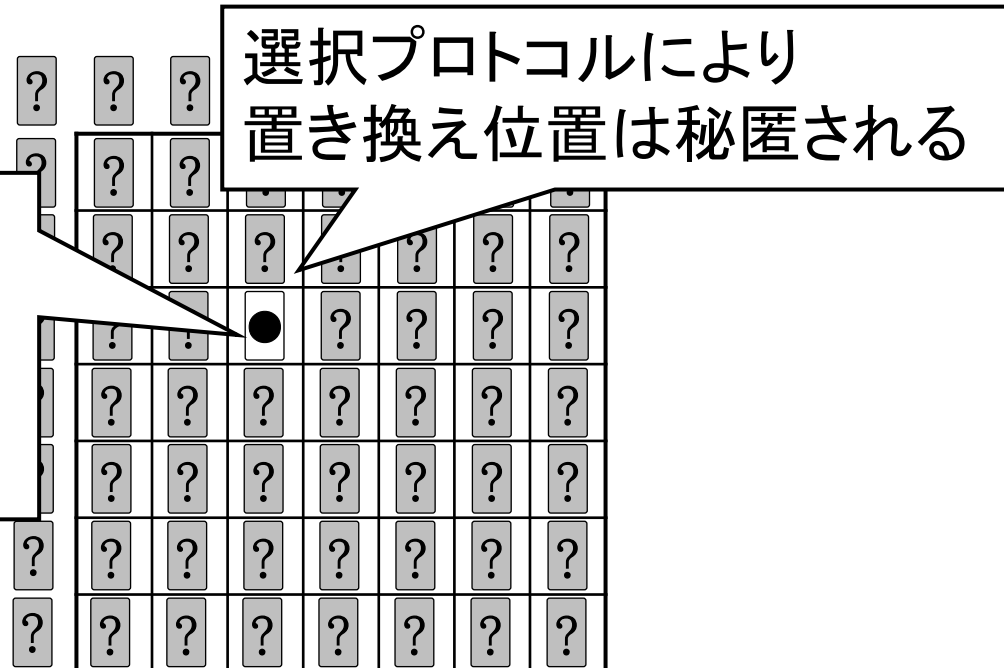
- 選択されたカードをめくり, 白カードであることを確認
- $\sigma = \text{connected, tree}$ の場合は, 黒カードに置き換える



提案プロトコル(形状 σ)

1. 白カードを各セルに置き, 番兵カードを盤面の左側に置く
2. 選択プロトコルを用いて, 白カードを選択する

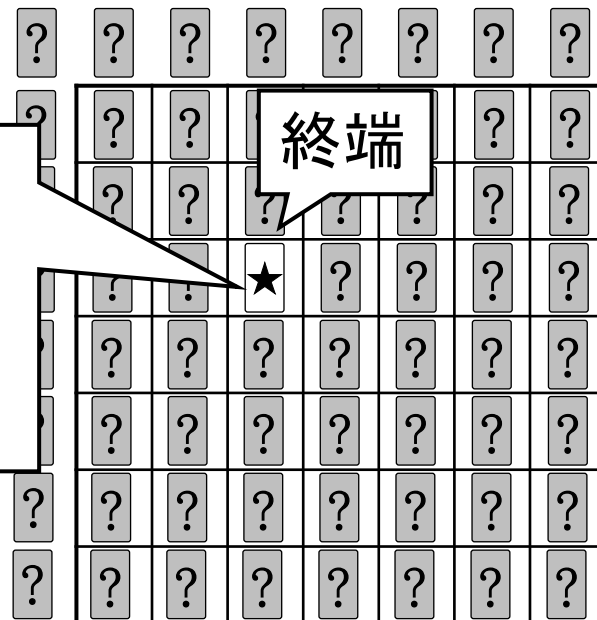
- 選択されたカードをめくり, 白カードであることを確認
- $\sigma = \text{connected, tree}$ の場合は, 黒カードに置き換える



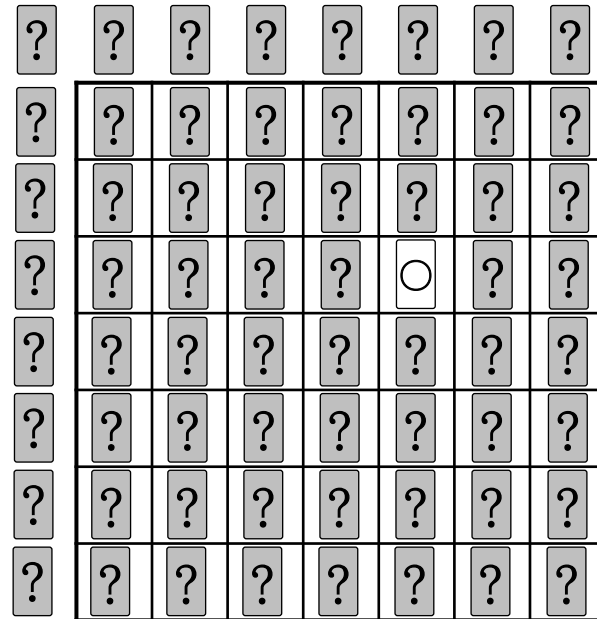
提案プロトコル(形状 σ)

1. 白カードを各セルに置き, 番兵カードを盤面の左上側に置く
2. 選択プロトコルを用いて, 白カードを選択する

- 選択されたカードをめくり, 白カードであることを確認
- $\sigma = \text{path, cycle}$ の場合は, 端カードに置き換える



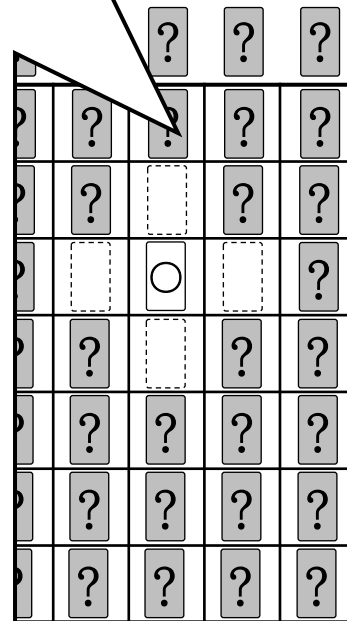
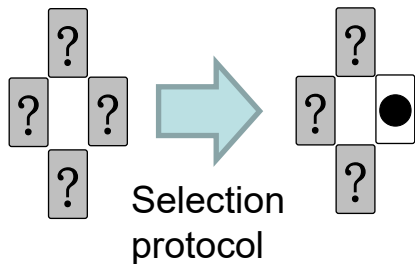
3. 選択プロトコルによる白カードの選択を $mn - 2$ 回繰り返す



3. 選択プロトコルによる白カードの選択を $mn - 2$ 回繰り返す

- 白カードの周囲4枚に対して, 2回目の選択プロトコルを行う
- $\sigma = \text{connected}$ の場合:
 - 2回目に選択されたカードが黒であることを確認する
 - 1回目に選択された白カードに対して, 3回目の選択プロトコルを行う※

2回目

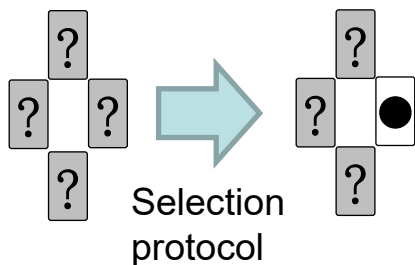


※ $mn - 2$ 回の最初のみ, 3回目の選択プロトコルは行わずに単に置き換える

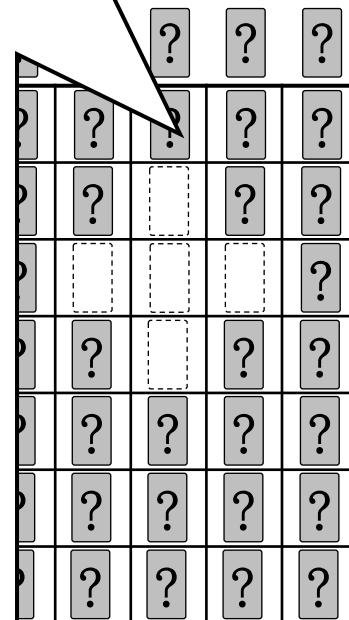
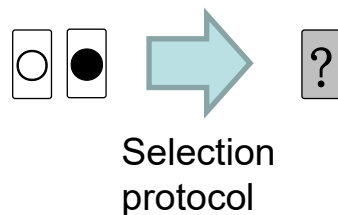
3. 選択プロトコルによる白カードの選択を $mn - 2$ 回繰り返す

- 白カードの周囲4枚に対して, 2回目の選択プロトコルを行う
- $\sigma = \text{connected}$ の場合:
 - 2回目に選択されたカードが黒であることを確認する
 - 1回目に選択された白カードに対して, 3回目の選択プロトコルを行う※

2回目



3回目

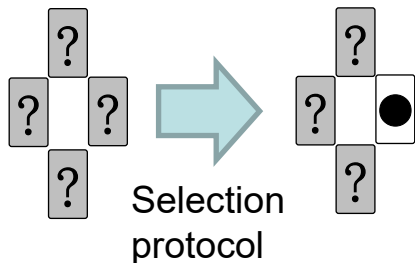


※ $mn - 2$ 回の最初のみ, 3回目の選択プロトコルは行わずに単に置き換える

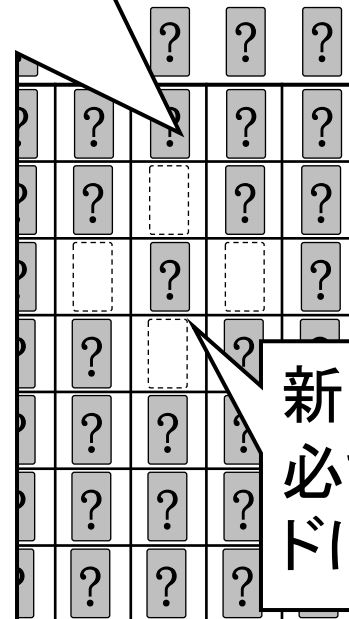
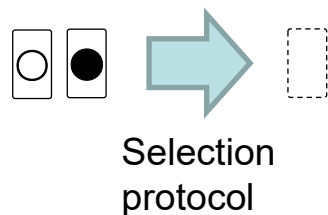
3. 選択プロトコルによる白カードの選択を $mn - 2$ 回繰り返す

- 白カードの周囲4枚に対して, 2回目の選択プロトコルを行う
- $\sigma = \text{connected}$ の場合:
 - 2回目に選択されたカードが黒であることを確認する
 - 1回目に選択された白カードに対して, 3回目の選択プロトコルを行う※
 - 選択されたカードを置く

2回目



3回目



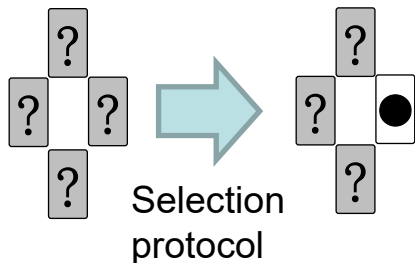
新しい黒カードは,
必ず既存の黒カード
に隣接する

※ $mn - 2$ 回の最初のみ, 3回目の選択プロトコルは行わずに単に置き換える

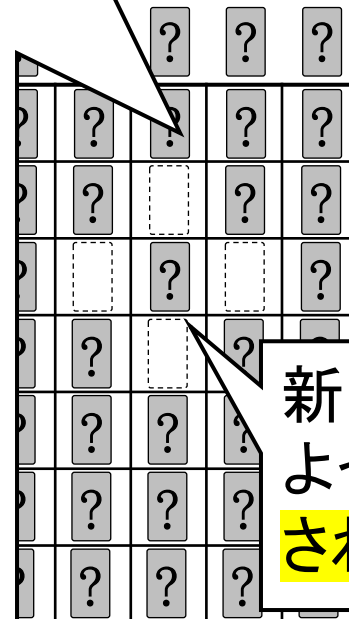
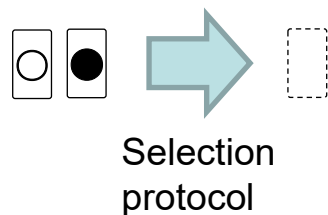
3. 選択プロトコルによる白カードの選択を $mn - 2$ 回繰り返す

- 白カードの周囲4枚に対して, 2回目の選択プロトコルを行う
- $\sigma = \text{tree}$ の場合:
 - 2回目に選択されたカードが黒, 他3枚が黒でないことを確認する
 - 1回目に選択された白カードに対して, 3回目の選択プロトコルを行う※
 - 選択されたカードを置く

2回目



3回目



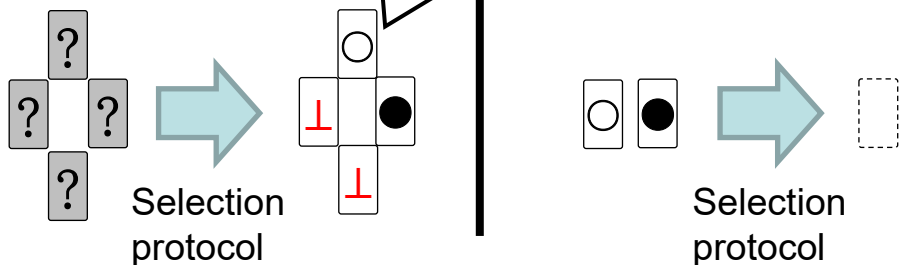
新しい黒カードによってcycleは形成されない

※ $mn - 2$ 回の最初のみ, 3回目の選択プロトコルは行わずに単に置き換える

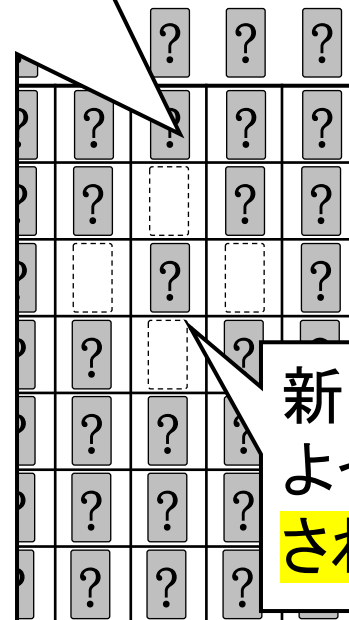
3. 選択プロトコルによる白カードの選択を $mn - 2$ 回繰り返す

- 白カードの周囲4枚に対して, 2回目の選択プロトコルを行う
- $\sigma = \text{tree}$ の場合:
 - 2回目に選択されたカードが黒, 他3枚が黒でないことを確認する
 - 1回目に選択された白カードに対して, 3回目の選択プロトコルを行う※
 - 選択されたカードを置く

2回目



番兵カードの数を
秘匿する必要あり



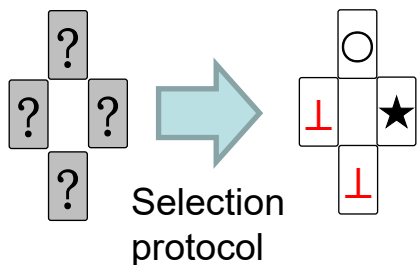
新しい黒カードに
よってcycleは形成
されない

※ $mn - 2$ 回の最初のみ, 3回目の選択プロトコルは行わずに単に置き換える

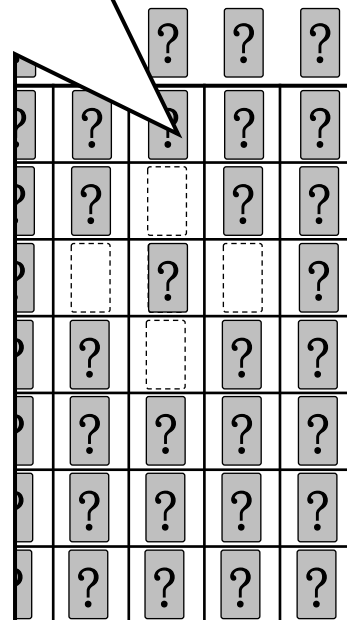
3. 選択プロトコルによる白カードの選択を $mn - 2$ 回繰り返す

- 白カードの周囲4枚に対して, 2回目の選択プロトコルを行う
- $\sigma = \text{path, cycle}$ の場合:
 - 2回目に選択されたカードが**端**, 他3枚が黒でないことを確認する
 - 1回目に選択された白カードと**2回目に選択された端カード**に対して, 3回目の選択プロトコルを行う※
 - 選択されたカードを置く

2回目



3回目



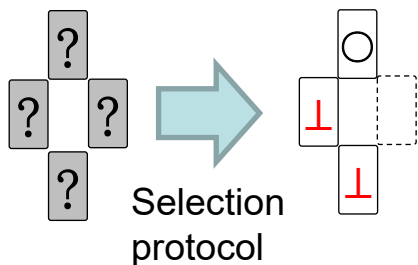
※ $mn - 2$ 回の最初のみ, 3回目の選択プロトコルは行わずに:

- $\sigma = \text{path}$: 1回目に選択された白カードを端カード, 2回目に選択された端カードを黒カードに置き換え
- $\sigma = \text{cycle}$: 1回目に選択された白カードを端カードに置き換え

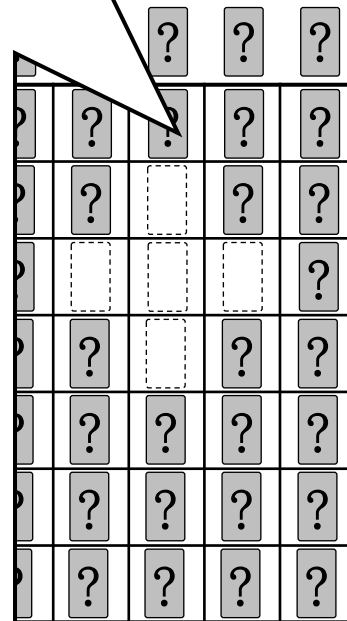
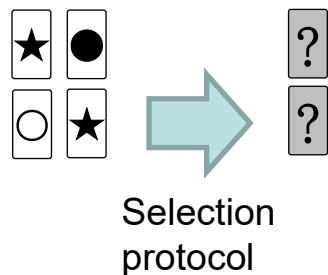
3. 選択プロトコルによる白カードの選択を $mn - 2$ 回繰り返す

- 白カードの周囲4枚に対して, 2回目の選択プロトコルを行う
- $\sigma = \text{path, cycle}$ の場合:
 - 2回目に選択されたカードが**端**, 他3枚が黒でないことを確認する
 - 1回目に選択された白カードと**2回目に選択された端カード**に対して, 3回目の選択プロトコルを行う※
 - 選択されたカードを置く

2回目



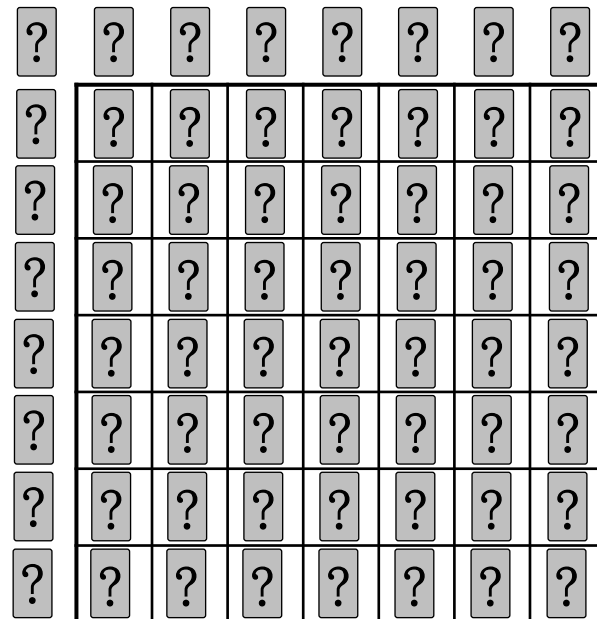
3回目



※ $mn - 2$ 回の最初のみ, 3回目の選択プロトコルは行わずに:

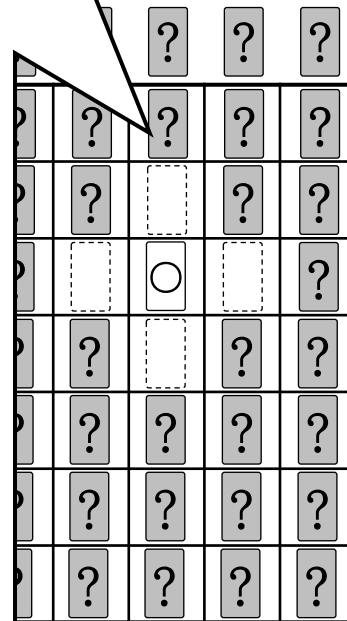
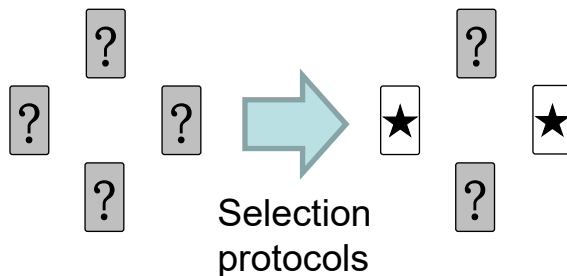
- $\sigma = \text{path}$: 1回目に選択された白カードを端カード, 2回目に選択された端カードを黒カードに置き換え
- $\sigma = \text{cycle}$: 1回目に選択された白カードを端カードに置き換え

4. 最後に, 同様に選択プロトコルを行う



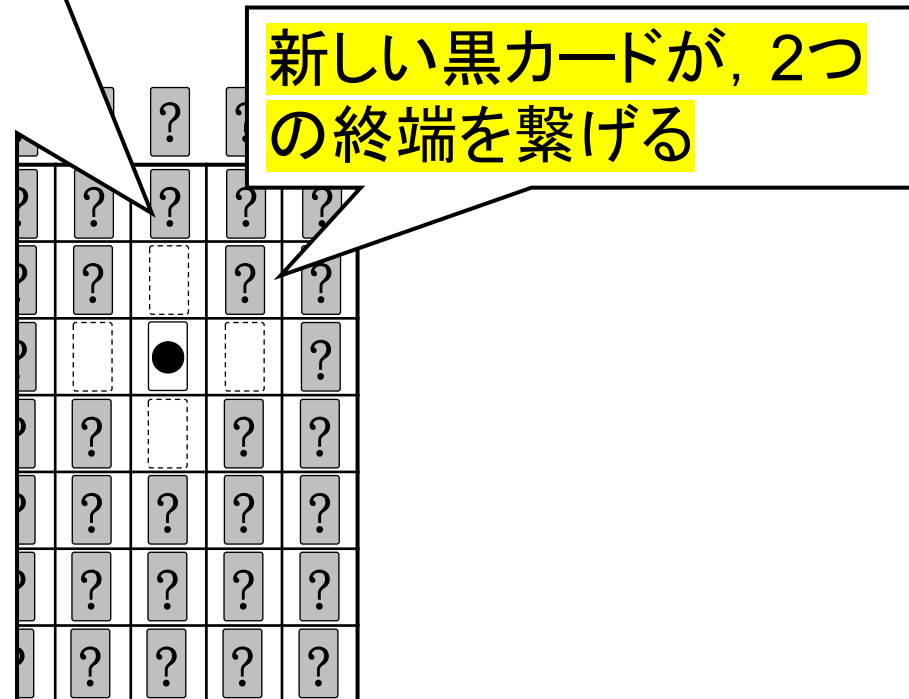
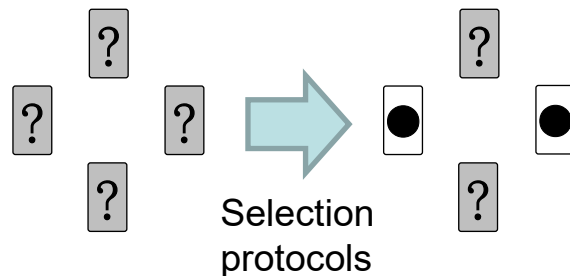
4. 最後に, 同様に選択プロトコルを1回行う

- $\sigma = \text{cycle}$ の場合のみ, 隣接する4枚に対して選択プロトコルを2回行う
 - 選択された2枚のカードが端であることを確認し, 黒に置き換え
 - 他2枚が黒でないことを確認
- 1回目に選択された白カード, 2,3回目に選択された端カードを黒に置き換える



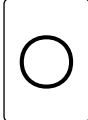
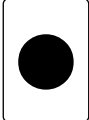
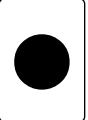
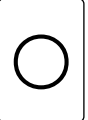
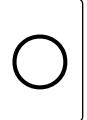
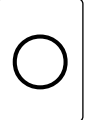
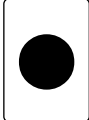
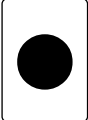
4. 最後に, 同様に選択プロトコルを1回行う

- $\sigma = \text{cycle}$ の場合のみ, 隣接する4枚に対して選択プロトコルを2回行う
 - 選択された2枚のカードが端であることを確認し, 黒に置き換え
 - 他2枚が黒でないことを確認
- 1回目に選択された白カード, 2,3回目に選択された端カードを黒に置き換える

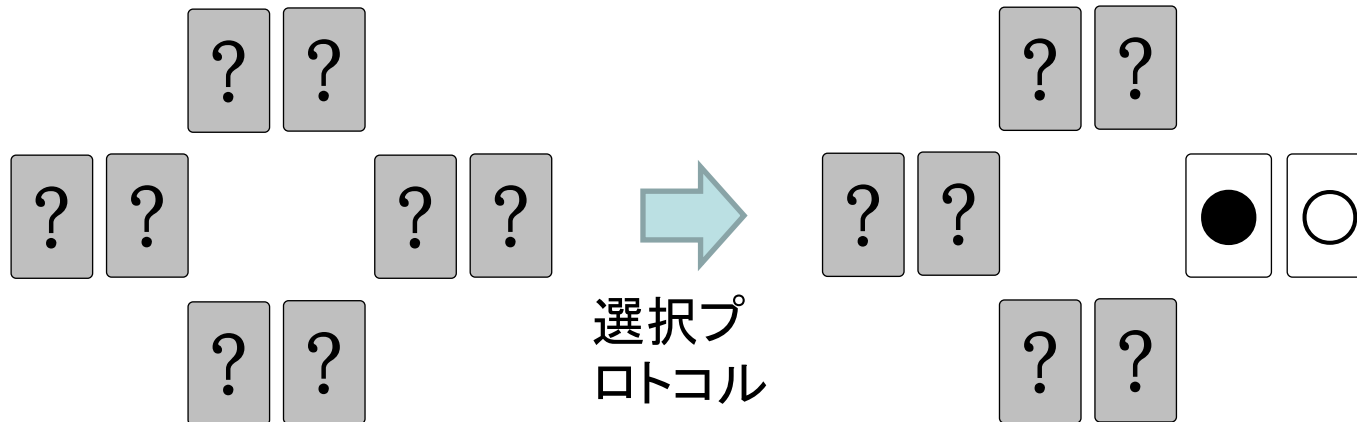


黒でないことを検証する方法

- 符号化を2枚に変更する:

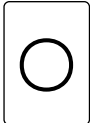
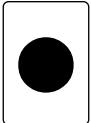
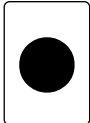
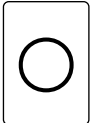
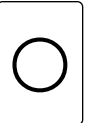
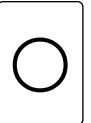
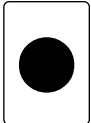
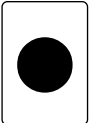
白:  , 黒:  , 番兵:  , 端:  

- 2枚のうち、左側のカードのみをめくればよい

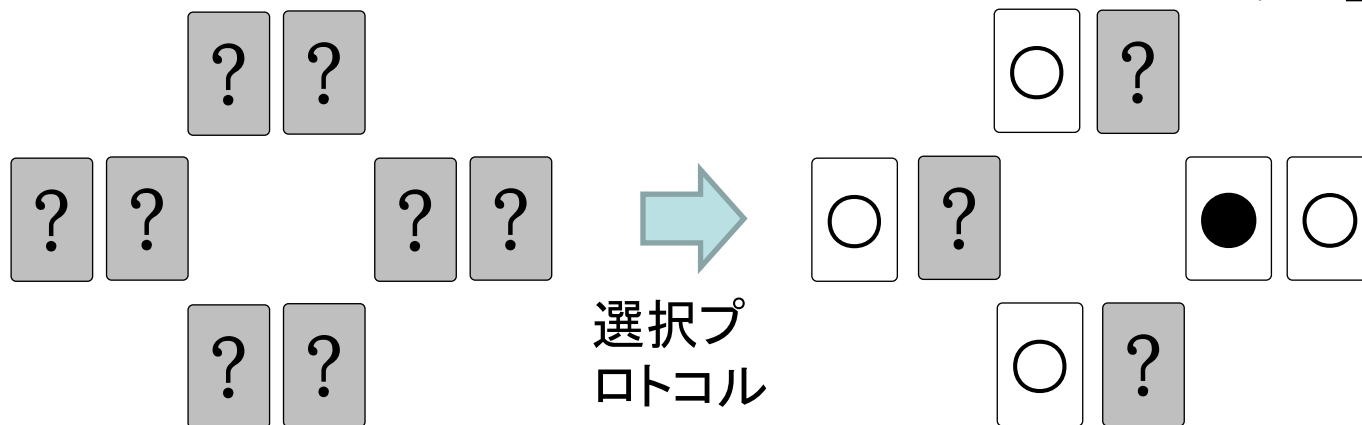


黒でないことを検証する方法

- 符号化を2枚に変更する:

白:  , 黒:  , 番兵:  , 端:  

- 2枚のうち, 左側のカードのみをめくればよい



白もしくは番兵, つまり
番兵の数は秘匿

1. Introduction
2. Preliminaries
3. Proposed Protocol
4. Application to Nurimeizu
5. Conclusion

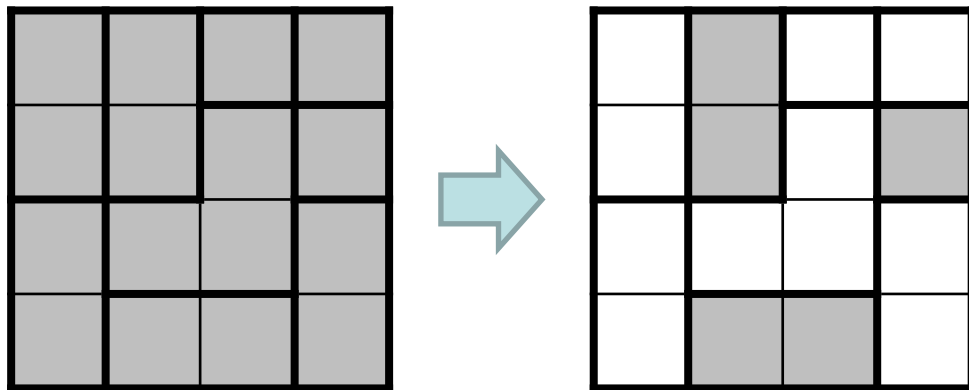
- 白マス黒く塗り, SからGまでを通ることができる迷路を作る
 - 太線で区切られた各部屋は, すべて黒マスかすべて白マスかのどちらかになる
 - すべての白マスはひとつながりになるが, 輪っかはない(つまり木)
 - S,G, Δ , $\bigcirc$ は白マス. すべての $\bigcirc$ はSからGの最短経路上, すべての Δ は最短経路から外れる
 - 白マスも黒マスも, 2×2 以上の領域にならない
- NP完全であることが証明されている [Iwamoto and Ide 2022]

ぬりめいず

多くのパズルに登場,
既存ZKPあり

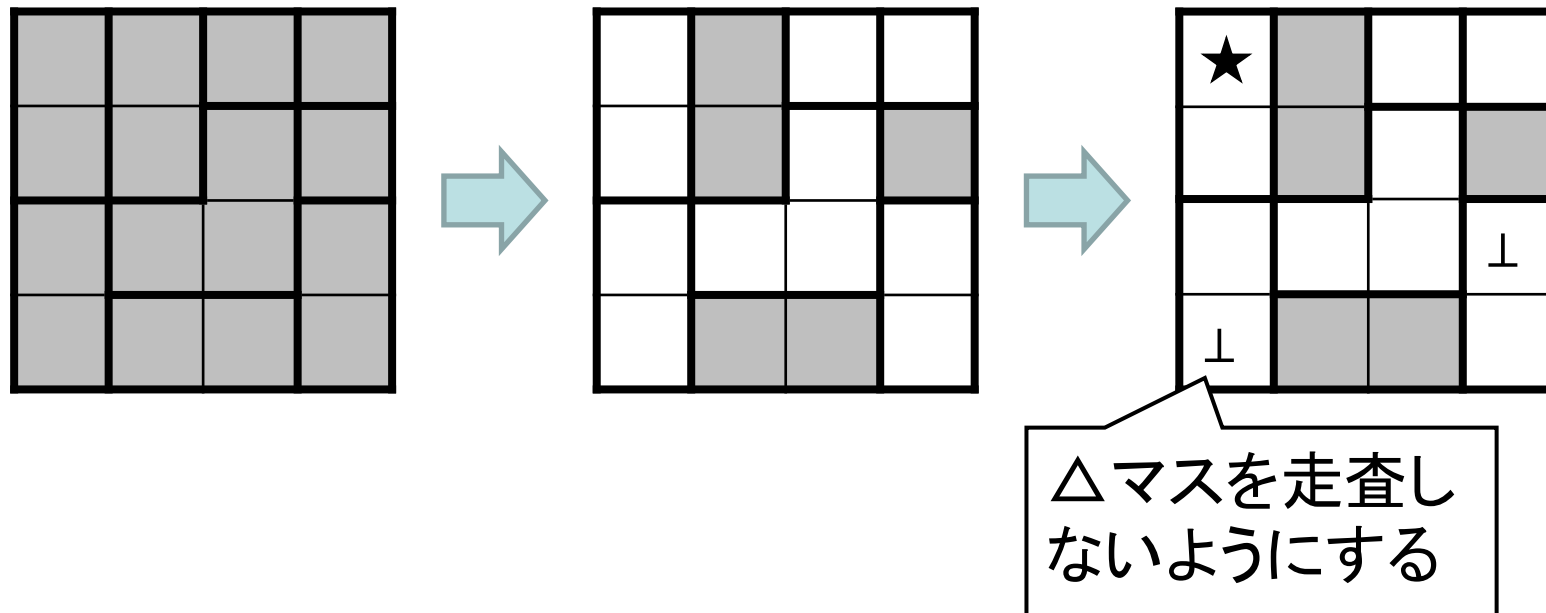
- 白マス黒く塗り, SからGまでを通ることができる迷路を作る
 - 太線で区切られた各部屋は, すべて黒マスかすべて白マスかのどちらかになる
 - すべての白マスはひとつながりになるが, 輪っかはない(つまり木)
 - S,G,△,○は白マス. すべての○はSからGの最短経路上, すべての△は最短経路から外れる
 - 白マスも黒マスも, 2×2 以上の領域にならない
- NP完全であることが証明されている [Iwamoto and Ide 2022]

1. すべてのマスが黒の状態からサブプロトコル(tree)を行い, 白マスを構築する



ぬりめいずに対するZKPの外観

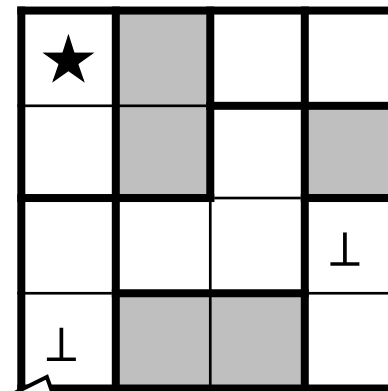
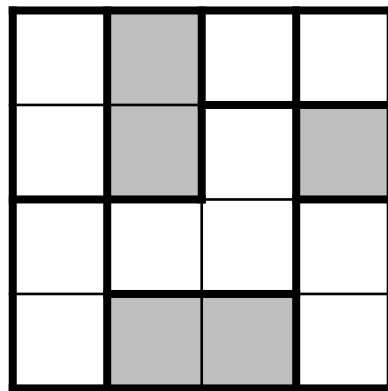
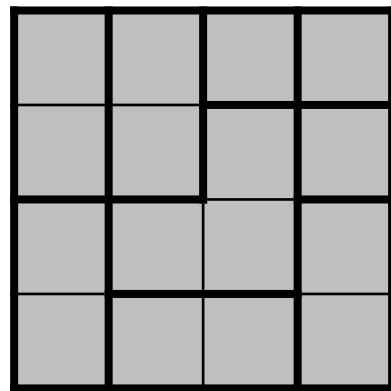
1. すべてのマスが黒の状態からサブプロトコル(tree)を行い, 白マスを構築する
2. 各△マスを番兵⊥, Sマスを端★に置き換える



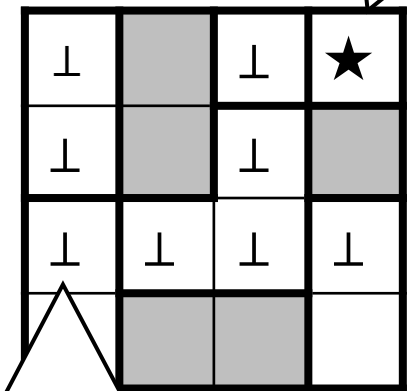
ぬりめいずに対するZKPの外観

1. すべてのマスが黒の状態からサブプロトコル(tree)を行い、白マスを構築する
2. 各△マスを番兵⊥, Sマスを端★に置き換える
3. サブプロトコル(path)を行い、番兵のpathを構築する

※ただし端カード→番兵カード
への置き換えは行わない



△マスを走査し
ないようにする



Gマスに置かれた
カードをめくり、端
であることを確認

○マスに置かれた
カードをめくり、番
兵であることを確認

1. Introduction
2. Preliminaries
3. Proposed Protocol
4. Application to Nurimeizu
5. Conclusion

■ **Table 1** Connectivity-based classification of Nikoli puzzles. Refer to [39] for their computational complexity.

Category	Structure	Representative Nikoli Puzzles
Single Tree	Tree	Nurimeizu (white)
Single Path	Path	Nurimeizu (goal)
Single Loop	Cycle	Dotchi-Loop, Masyu [18], Moon-or-Sun [8], Mid-loop, Nagareru Slitherlink [18], Suraromu, Yajilin
Single Connected Region	Polyomino	Heyawake [29], Hitori [29], Kurdoko [30], Kurotto, LITS, Nurikabe (black) [29], Nurimisaki [30], Nuritwin, Sukoro [35], Yajisan Kazusan
Multiple Connected Regions	Polyominoes	Evolomino, Fillomino, LITS (tetrominoes), Nurikabe (white) [29], Ripple Effect [32], Sashigane Shakashaka [22], Shikaku [34]
Multiple Paths	Disjoint open simple paths	Hebi, Herugolf [23], Hotaru Beam [28], Pencils
Connected Islands	Connected spanning subgraph	Hashiwokakero [33]
Chain	Corner-adjacent polyominoes	Chained Block

辺を描く場合
の統一的記述

連結領域が複数
個ある場合
の統一的記述