1回または2回のシャッフルに関する未解決問題

2025年5月28日(水)＠九州大学伊都キャンパス

品川 和雅（筑波大）

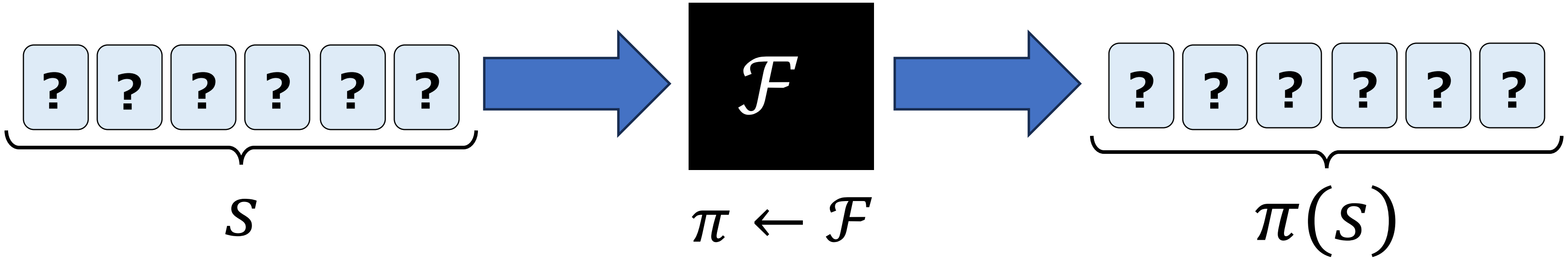
本講演の概要

1. シャッフルについて
2. MPCのシャッフル回数
3. ZKPのシャッフル回数
4. まとめ

1. シャッフルについて

シャッフルについて

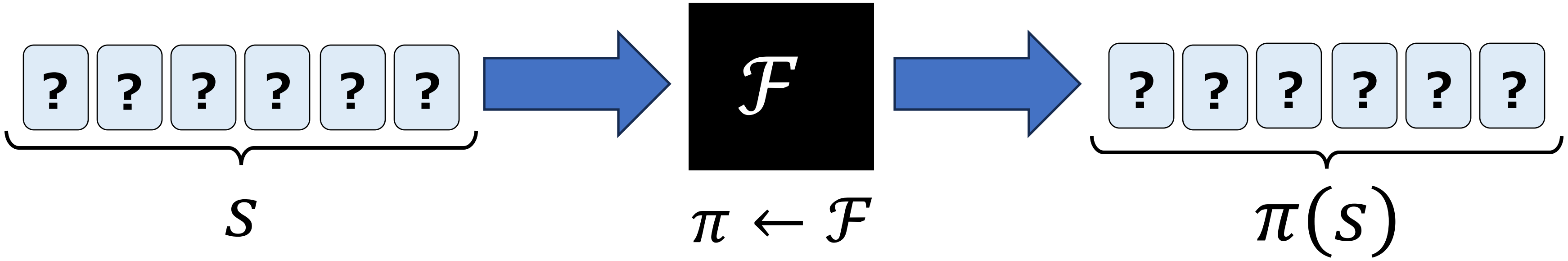
- n枚に対するシャッフルは、S_n上の確率分布$\mathcal{F}$で定まる以下の操作



- 実際は$\Pi := \mathrm{supp}(\mathcal{F}) = \{\pi \in S_n \mid \mathcal{F}(\pi) > 0\}$が重要であるため、シャッフルを$(\mathrm{shuffle}, \Pi, \mathcal{F})$と表記する
- 重要なシャッフルクラス
  - **閉シャッフル**：$\Pi := \mathrm{supp}(\mathcal{F})$が群
  - **一様シャッフル**：$\mathcal{F}$がΠ上の一様分布（このとき$(\mathrm{shuffle}, \Pi)$と書いてOK）
  - **一様閉シャッフル**：閉シャッフルかつ一様シャッフル

シャッフルについて

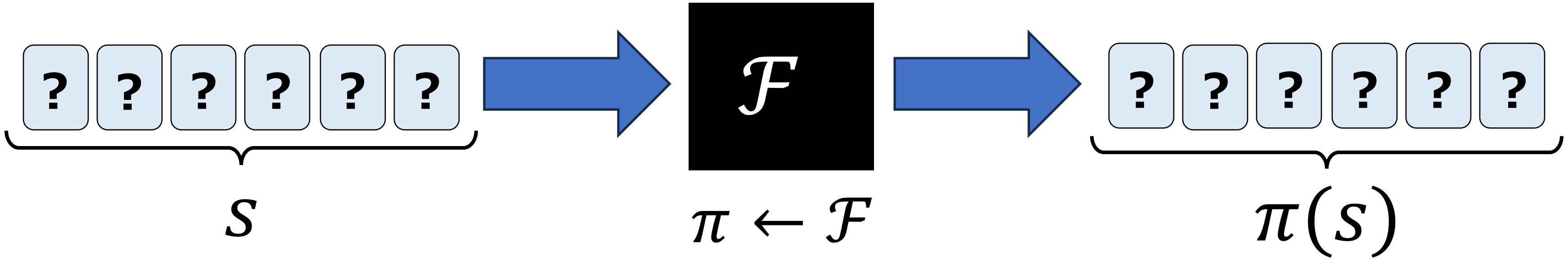
- n枚に対するシャッフルは、S_n上の確率分布$\mathcal{F}$で定まる以下の操作



- 実際は$\Pi \coloneqq \mathrm{supp}(\mathcal{F}) = \{\pi \in S_n \mid \mathcal{F}(\pi) > 0\}$が重要であるため、シャッフルを$(\mathrm{shuffle}, \Pi, \mathcal{F})$と表記する
- 重要なシャッフルクラス
  - **閉シャッフル**：$\Pi \coloneqq \mathrm{supp}(\mathcal{F})$が群
  - **一様シャッフル**：$\mathcal{F}$がΠ上の一様分布（このとき$(\mathrm{shuffle}, \Pi)$と書いてOK）
  - **一様閉シャッフル**：閉シャッフルかつ一様シャッフル

シャッフルについて

- n枚に対するシャッフルは、S_n上の確率分布$\mathcal{F}$で定まる以下の操作



- 実際は$\Pi \coloneqq \mathrm{supp}(\mathcal{F}) = \{\pi \in S_n \mid \mathcal{F}(\pi) > 0\}$が重要であるため、シャッフルを$(\mathrm{shuffle}, \Pi, \mathcal{F})$と表記する
- 重要なシャッフルクラス
  - **閉シャッフル**：$\Pi \coloneqq \mathrm{supp}(\mathcal{F})$が群
  - **一様シャッフル**：$\mathcal{F}$がΠ上の一様分布（このとき$(\mathrm{shuffle}, \Pi)$と書いてOK）
  - **一様閉シャッフル**：閉シャッフルかつ一様シャッフル

シャッフルの世界地図

実装容易

ランダムカット　完全シャッフル

ランダム二等分割カット

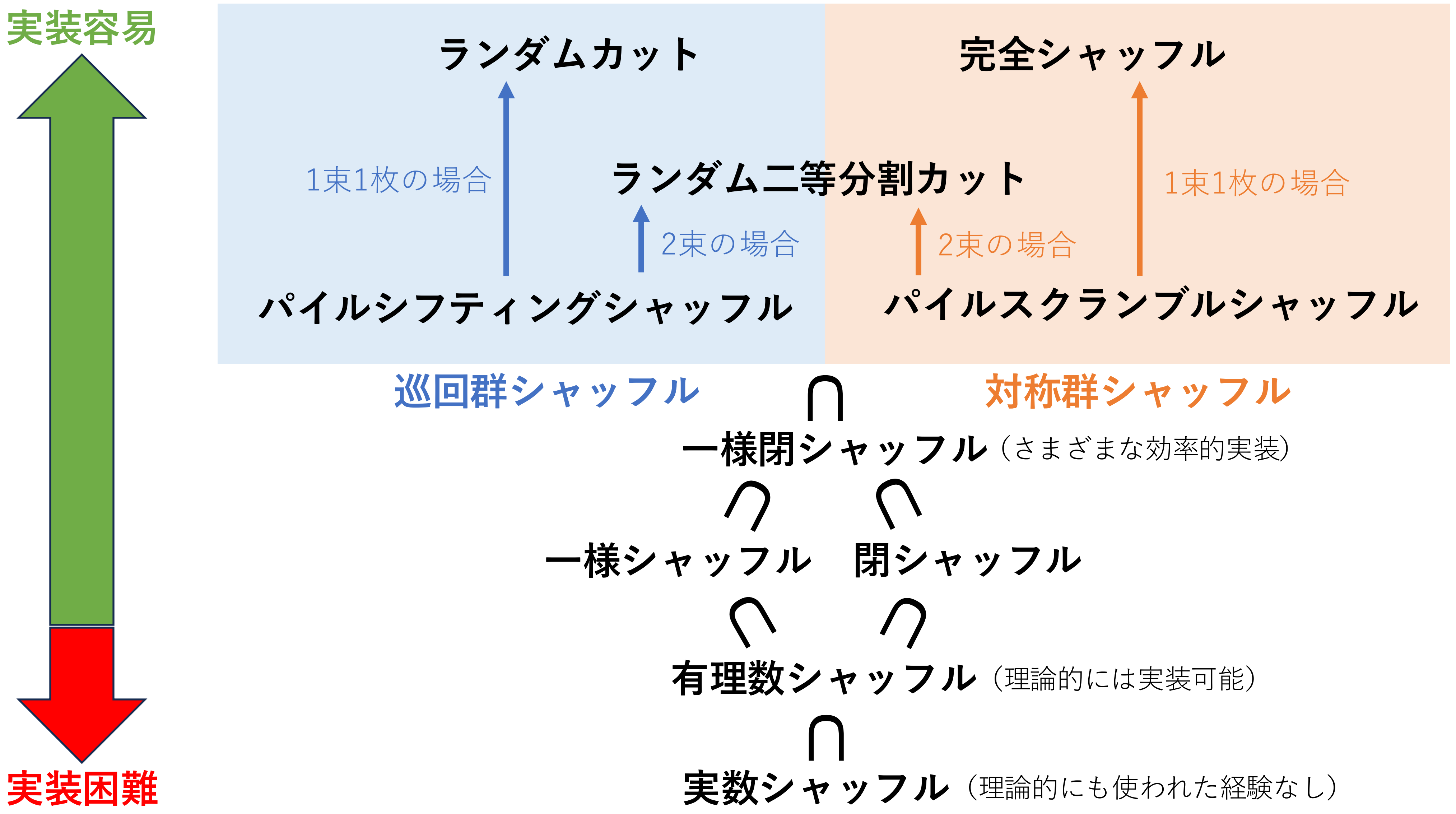
パイルシフティングシャッフル　パイルスクランブルシャッフル

実装困難

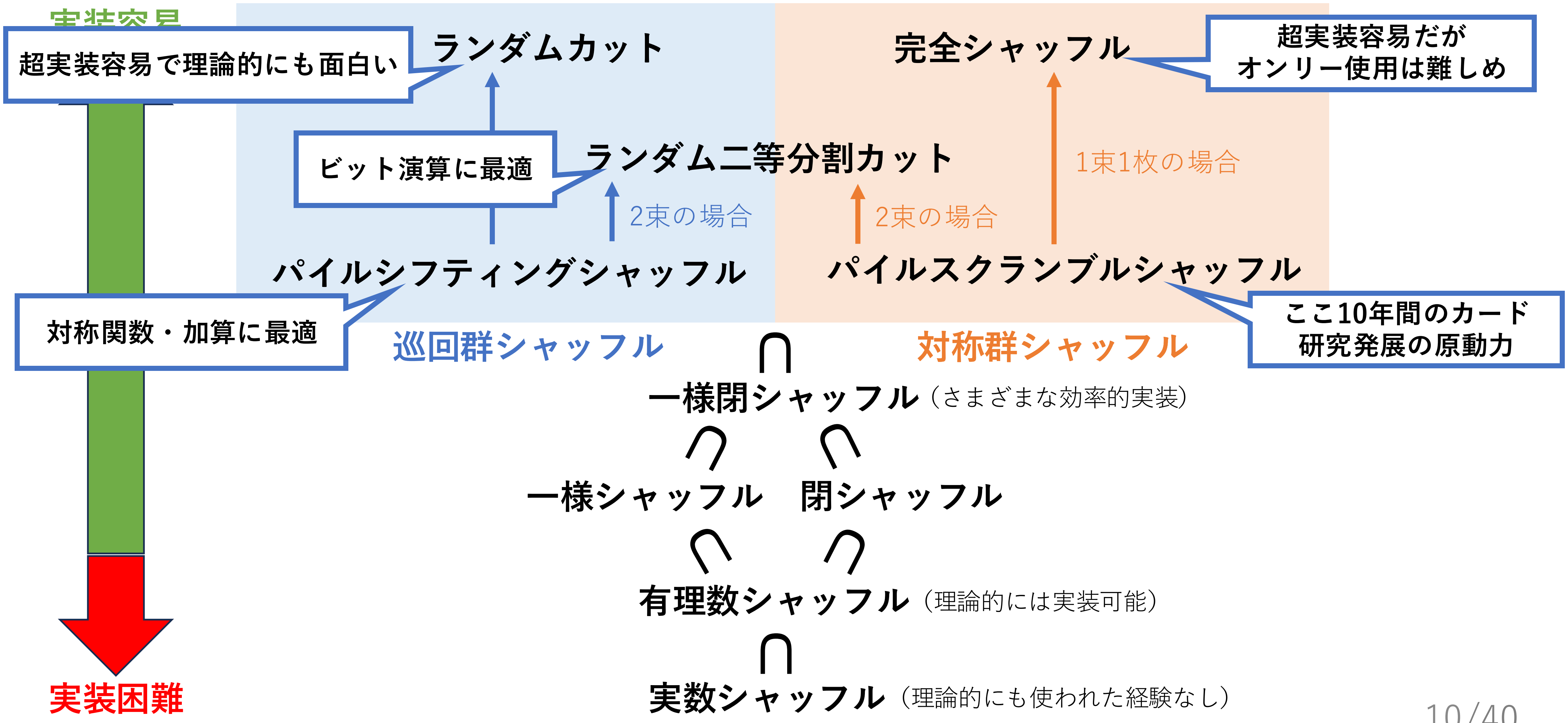
シャッフルの世界地図


実装容易
ランダムカット
完全シャッフル
1束1枚の場合
ランダム二等分割カット
1束1枚の場合
2束の場合
2束の場合
パイルシフティングシャッフル
パイルスクランブルシャッフル
巡回群シャッフル
対称群シャッフル
実装困難

シャッフルの世界地図


実装容易
ランダムカット
完全シャッフル
1束1枚の場合
ランダム二等分割カット
1束1枚の場合
2束の場合
2束の場合
パイルシフティングシャッフル
パイルスクランブルシャッフル
巡回群シャッフル
∩
対称群シャッフル
一様閉シャッフル（さまざまな効率的実装）
∩
∩
一様シャッフル
閉シャッフル
∩
∩
有理数シャッフル（理論的には実装可能）
∩
実数シャッフル（理論的にも使われた経験なし）
実装困難

シャッフルの世界地図

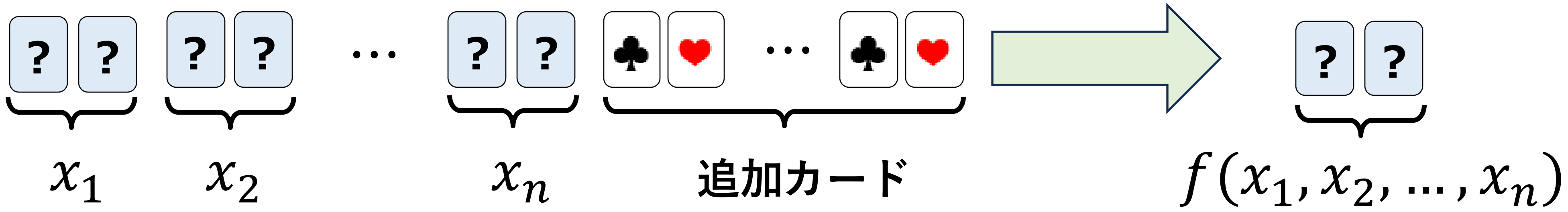


超実装容易で理論的にも面白い
ランダムカット
完全シャッフル
超実装容易だが
オンリー使用は難しめ
ビット演算に最適
ランダム二等分割カット
1束1枚の場合
2束の場合
2束の場合
パイルシフティングシャッフル
パイルスクランブルシャッフル
対称関数・加算に最適
ここ10年間のカード
研究発展の原動力
巡回群シャッフル
対称群シャッフル
∩
一様閉シャッフル（さまざまな効率的実装）
一様シャッフル
閉シャッフル
有理数シャッフル（理論的には実装可能）
∩
実数シャッフル（理論的にも使われた経験なし）
実装困難

2. MPCのシャッフル回数

MPCの問題設定

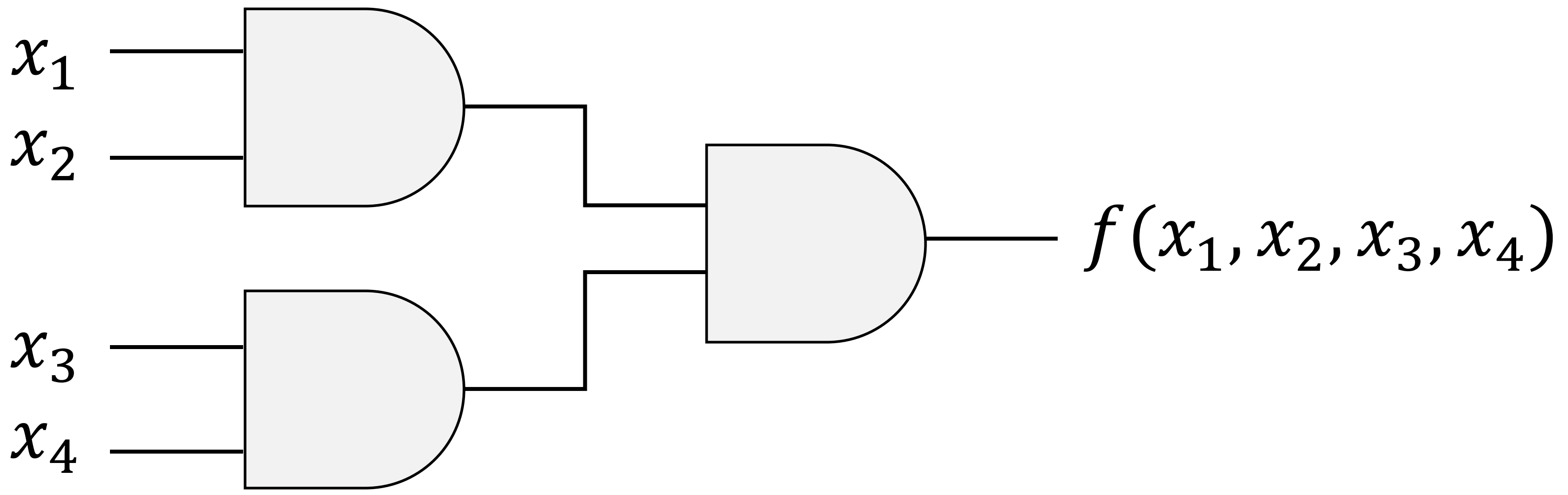
- 関数$f\colon \{0,1\}^n \rightarrow \{0,1\}$に対するMPCプロトコル



- 入力コミットメント（$2n$枚）と追加カードを用いる
- シャッフル回数はどこまで小さくできるか？

初期の結果：$|C|$回のRBC[MS09]

- 回路Cのゲート数を$|C|$と表すことにする
- Mizuki-SoneのAND/XOR/COPYプロトコルを順次適用→$|C|$回
  - 各プロトコルはRBC 1回



[MS09] Mizuki and Sone. Six-Card Secure AND and Four-Card Secure XOR. FAW 2009.

初期の結果：$\boldsymbol{n}$回のRBC[SMS+15]

- 任意関数f: $\{0,1\}^n \rightarrow \{0,1\}$に対するRBC n回のプロトコル

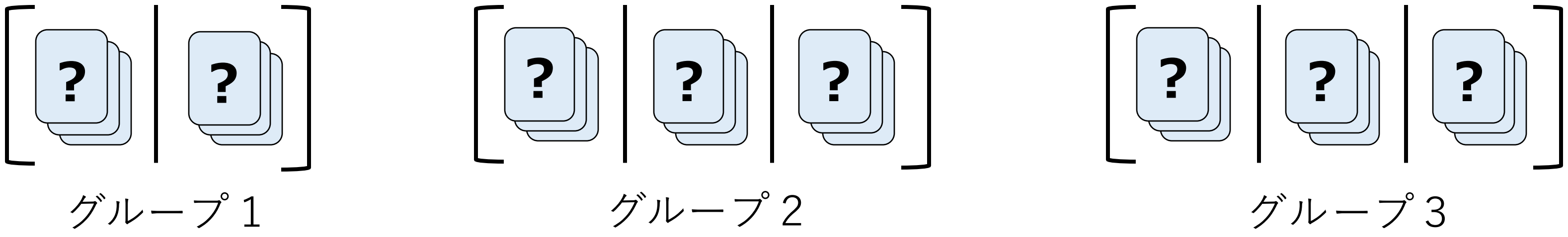
- プロトコル手順の概要
  1. 全ての出力値（2^n通り）のコミットメントを作成する
  2. 各入力x_iを用いて、2つの候補から1つを選択（各RBC 1回）

- 追加カード枚数は2^{n+1}枚

[SMS+15] Shinagawa, Mizuki, Schuldt, Nuida, Kanayama, Nishide, Hanaoka, Okamoto. Multi-Party Computation with Small Shuffle Complexity Using Regular Polygon Cards. ProvSec 2015.

初の定数回：2回のPSS [品川ら17]

- **バッチング技術**
  - 並列的に複数回のPSS（RBC含む）を実行したいとする

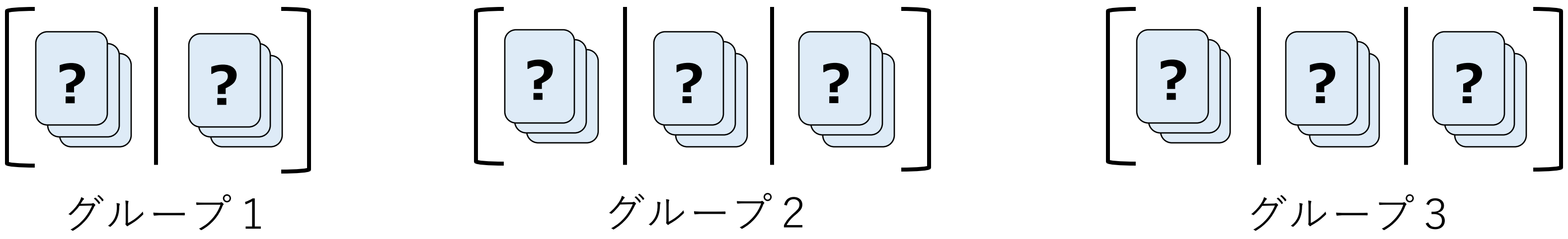


  - 数字カードを付加し、PSSを1回実行し、数字カードをめくればよい

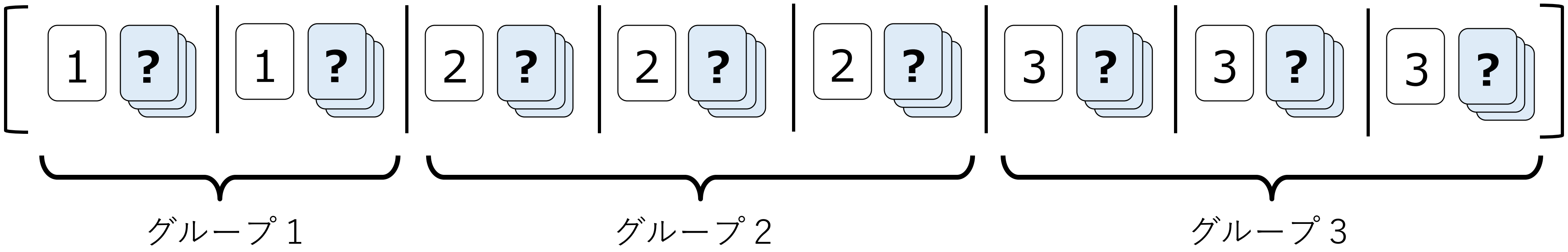
[品川ら17] 品川, 縫田, 西出, 花岡. 任意の論理関数を計算するシャッフル2 回のカードベースプロトコル. SCIS 2017.

初の定数回：2回のPSS [品川ら17]

- **バッチング技術**
  - 並列的に複数回のPSS（RBC含む）を実行したいとする



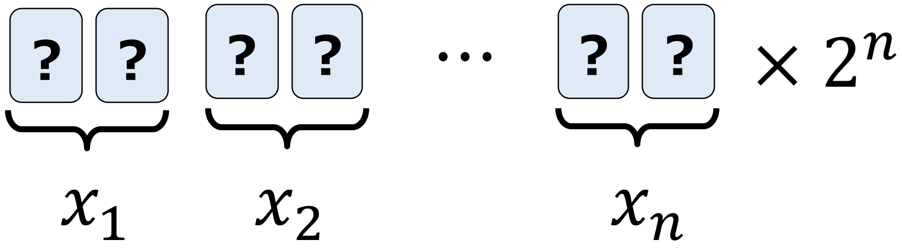
  - 数字カードを付加し、PSSを1回実行し、数字カードをめくればよい



[品川ら17] 品川, 縫田, 西出, 花岡. 任意の論理関数を計算するシャッフル2 回のカードベースプロトコル. SCIS 2017.

初の定数回：2回のPSS [品川ら17]

1. 入力$x_1, x_2, \dots, x_n$をそれぞれ2^n個コピーする（PSS 1回）

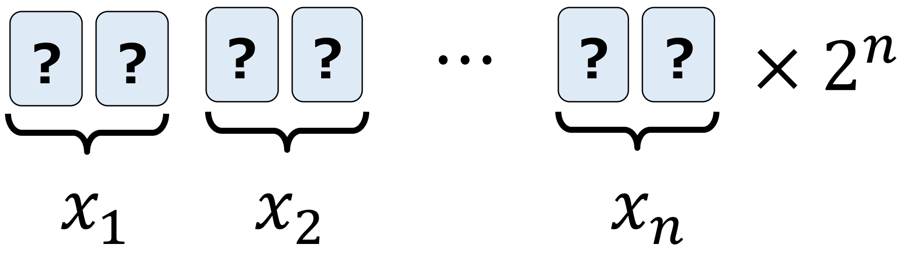


2. 各$\ell = (\ell_1, \ell_2, \dots, \ell_n) \in \{0,1\}^n$に対して以下のカード列を作る

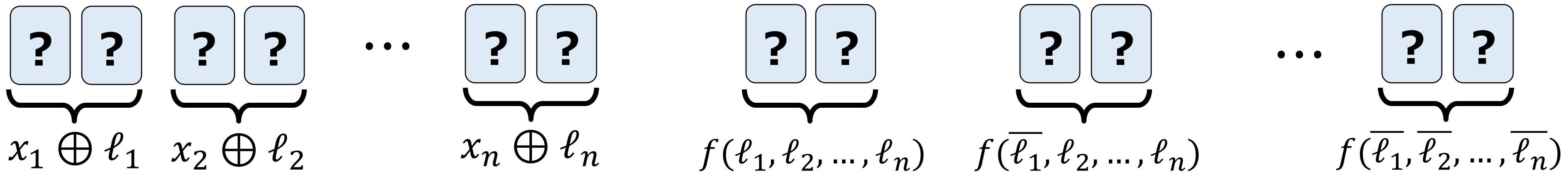
[品川ら17] 品川, 縫田, 西出, 花岡. 任意の論理関数を計算するシャッフル2 回のカードベースプロトコル. SCIS 2017.

初の定数回：2回のPSS [品川ら17]

1. 入力$x_1, x_2, \ldots, x_n$をそれぞれ2^n個コピーする（PSS 1回）



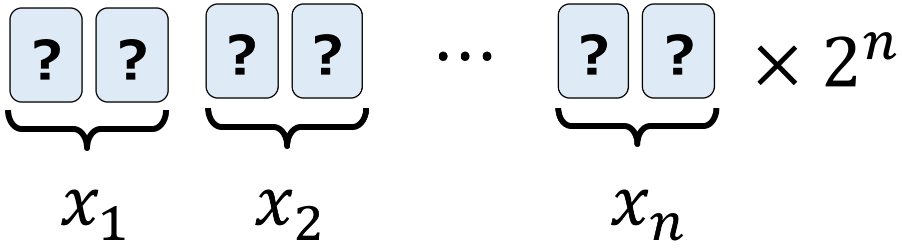
2. 各$\ell = (\ell_1, \ell_2, \ldots, \ell_n) \in \{0,1\}^n$に対して以下のカード列を作る



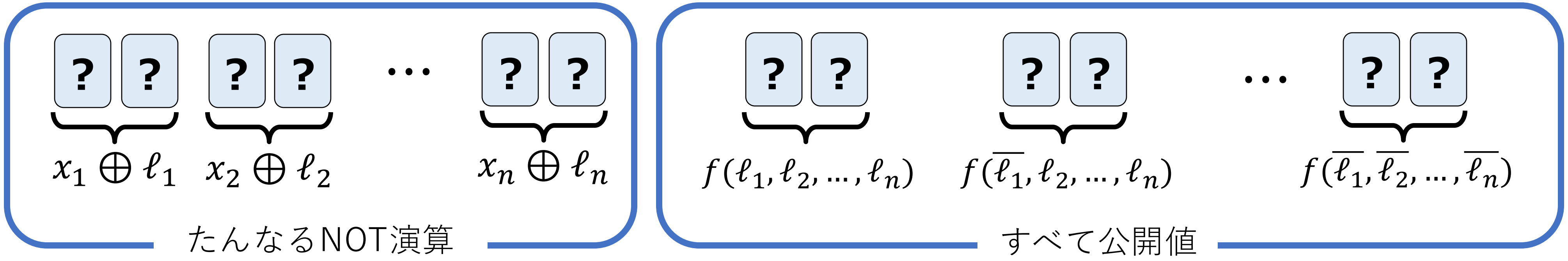
[品川ら17] 品川, 縫田, 西出, 花岡. 任意の論理関数を計算するシャッフル2 回のカードベースプロトコル. SCIS 2017.

初の定数回：2回のPSS [品川ら17]

1. 入力$x_1, x_2, \dots, x_n$をそれぞれ2^n個コピーする（PSS 1回）



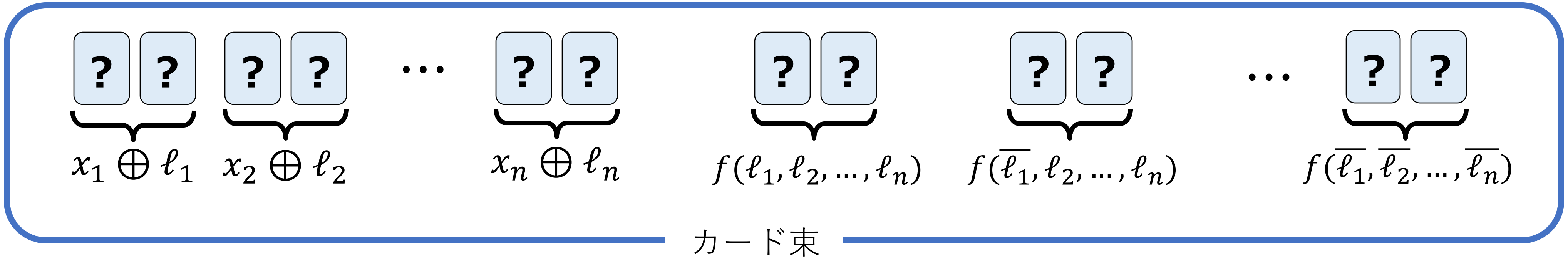
2. 各$\ell = (\ell_1, \ell_2, \dots, \ell_n) \in \{0,1\}^n$に対して以下のカード列を作る



[品川ら17] 品川, 縫田, 西出, 花岡. 任意の論理関数を計算するシャッフル2 回のカードベースプロトコル. SCIS 2017.

初の定数回：2回のPSS [品川ら17]

3. 各$\ell \in \{0,1\}^n$のカード列を束として、2^n個の束のPSSを実行

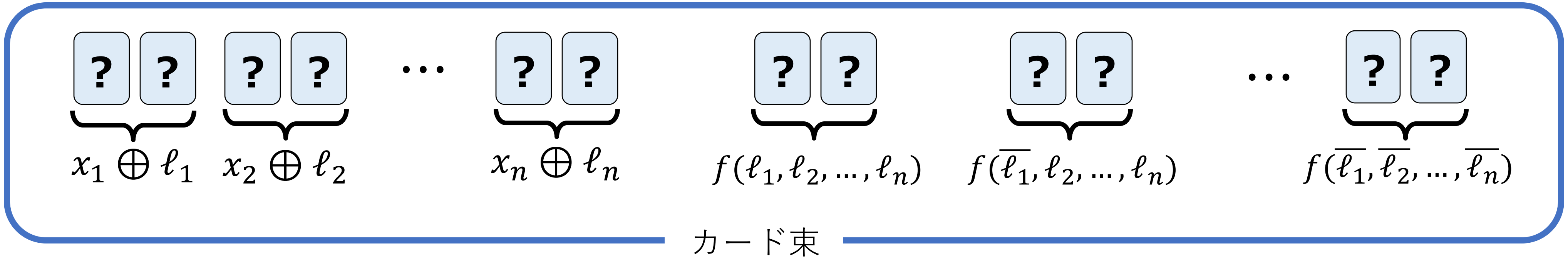


4. 束を1つ選び、$(x_1 \oplus \ell_1, \dots, x_n \oplus \ell_n)$を公開し、該当箇所を出力

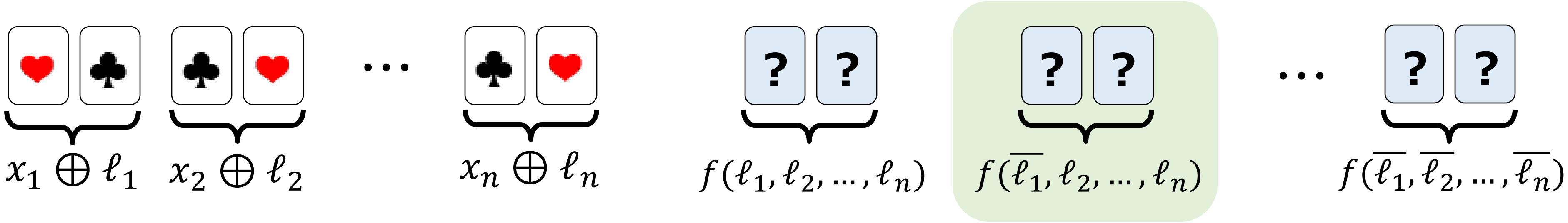
[品川ら17] 品川, 縫田, 西出, 花岡. 任意の論理関数を計算するシャッフル2 回のカードベースプロトコル. SCIS 2017.

初の定数回：2回のPSS [品川ら17]

3. 各$\ell \in \{0,1\}^n$のカード列を束として、2^n個の束のPSSを実行

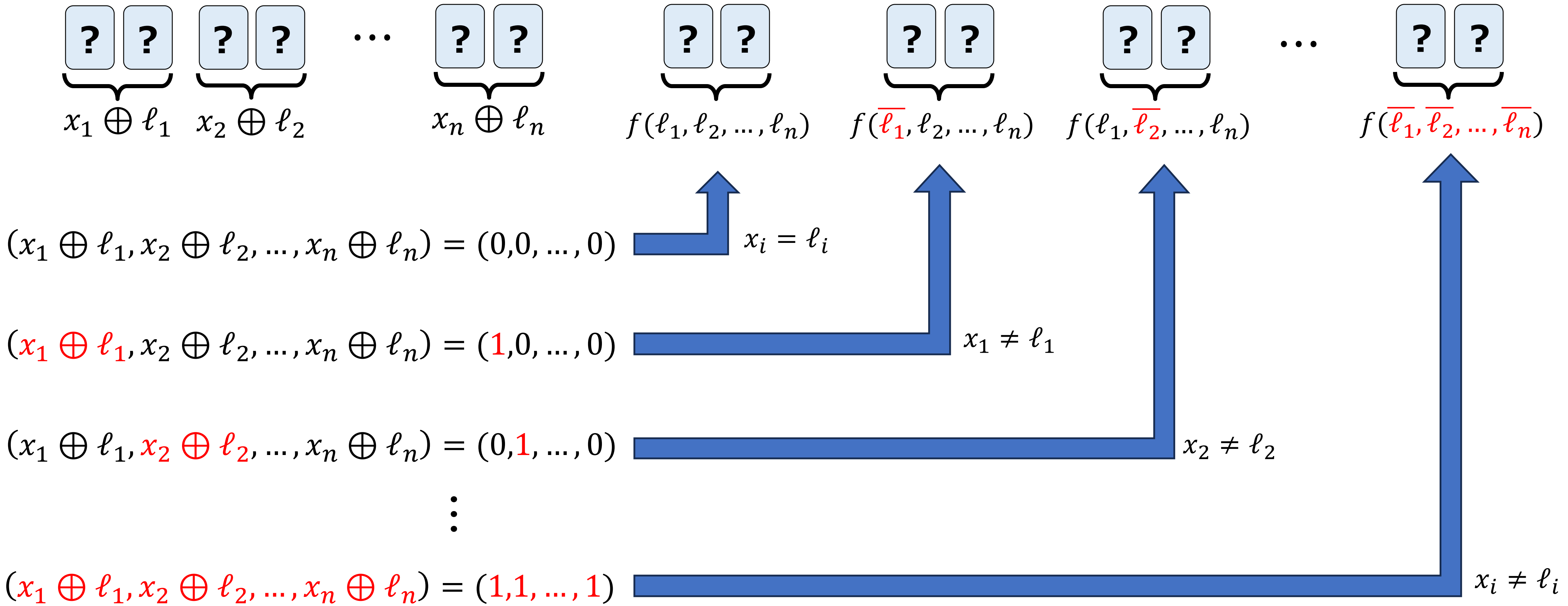


4. 束を1つ選び、$(x_1 \oplus \ell_1, \dots, x_n \oplus \ell_n)$を公開し、該当箇所を出力



[品川ら17] 品川, 縫田, 西出, 花岡. 任意の論理関数を計算するシャッフル2 回のカードベースプロトコル. SCIS 2017.

初の定数回：2回のPSS [品川ら17]



[品川ら17] 品川, 縫田, 西出, 花岡. 任意の論理関数を計算するシャッフル2 回のカードベースプロトコル. SCIS 2017.

Garbled回路：**2**回のPSS [SN21]

- Garbled回路のアイディアを用いてプロトコルを実装する
    - ゲートランダマイズ：ゲート（真理値表）の行と列をデタラメにする
    - ワイヤランダマイズ：ワイヤの値を乱数でマスクする

- それぞれのランダマイズでは並列的なPSSがたくさん登場する

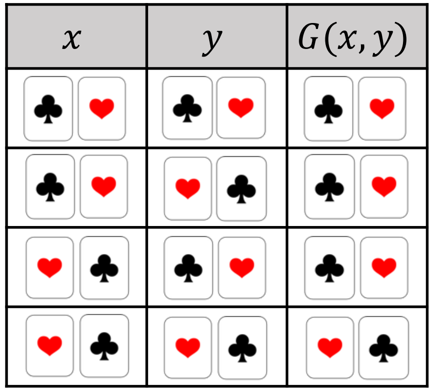
- バッチング技術を用いて**PSS 2回**に集約可能

- PSS 2回は連続適用のため、**一様閉シャッフル1回**ともみなせる

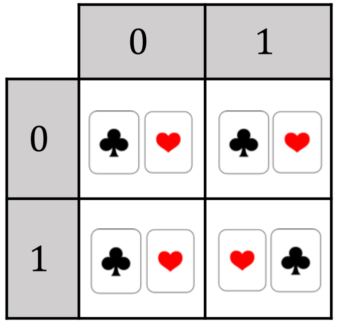
[SN21] Shinagawa, Nuida. A single shuffle is enough for secure card-based computation of any Boolean circuit. Discret. Appl. Math. 2021.

Garbled回路の効率化[TMM23], [OSN+23]

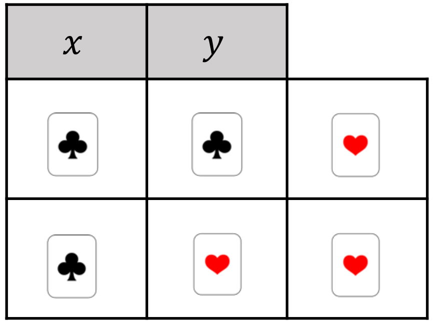
- Tozawaらの方式
  - 1ゲート8枚
  - 一様閉シャッフル1回
- Onoらの方式
  - 1ゲート6枚
  - 一様シャッフル1回



Shinagawa-Nuida (24枚)



Tozawaら(8枚)



Onoら(6枚)

- シャッフルの種類とカード枚数のトレードオフ
  - PSS　：Shinagawa-Nuidaの1ゲート24枚
  - 一様閉：Tozawa-Morita-Mizukiの1ゲート8枚
  - 一様　：Onoらの1ゲート6枚

[TMM23] Tozawa, Morita, Mizuki. Single-Shuffle Card-Based Protocol with Eight Cards per Gate. UCNC 2023.
[OSN+23] Ono, Shinagawa, Nakai, Watanabe, Iwamoto. Single-Shuffle Card-Based Protocols with Six Cards per Gate. ICISC 2023.

MPCのシャッフル回数に関する未解決問題

- 実現可能性に関する未解決問題
  - PSS 1回のプロトコルは構成可能か？

- ゲートあたりの枚数の未解決問題
  - PSS 2回の場合のカード枚数をゲート24枚より削減可能か？
  - 一様閉シャッフル1回の場合のカード枚数をゲート8枚より削減可能か？
  - 一様シャッフル1回の場合のカード枚数をゲート6枚より削減可能か？

- 他のシャッフルの場合
  - 巡回群シャッフルやパイルシフティングの場合の最小回数は？

3. ZKPのシャッフル回数

ゼロ知識証明（ZKP）の問題設定

- 証明者はあるパズルの答えを知っている
- 検証者は「このパズルには答えが存在しないのでは」と疑っている
- 検証者に「答えが存在すること」をどうやって納得させられるか？
- ただし、答え自体を教えて問題を解く楽しみを奪ってはいけない

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

証明者

検証者

カードベースZKP

- カードを用いてZKPを実現する研究分野
  - 物理ZKPとも呼ばれる（こちらはカード以外の道具も包含している）

- さまざまなパズル・問題に対するプロトコルが研究されてきた
  - パズルの例：数独、スリザーリンク、マカロ、四角に切れ、覆面算など
  - パズル以外：グラフ同型問題、三彩色問題、パンケーキソートなど

- **数独のZKP**のシャッフル回数についての研究進展を紹介する
  - 数独はクロスワードと並んで世界で最も有名なペンシルパズル
  - 多くの人はルールを知っており、ZKPの教育にも使いやすい

数独

問題

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

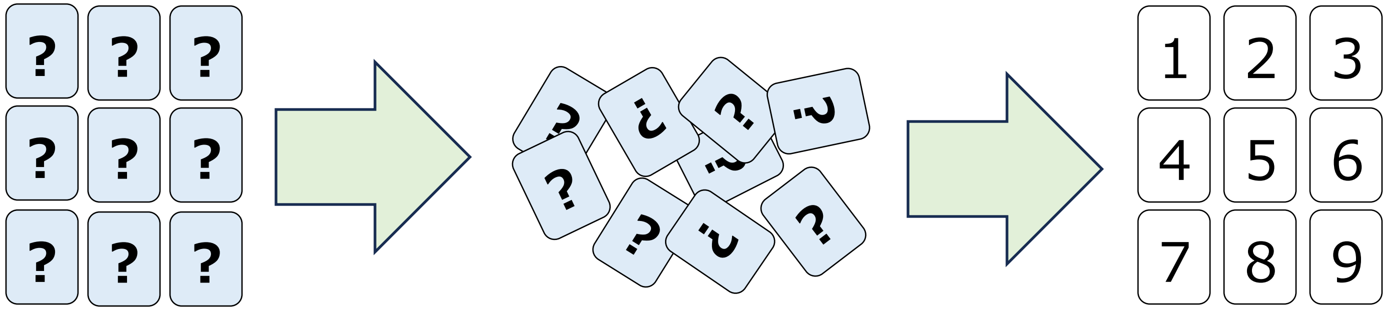
答え

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

- タテ・ヨコ・ブロックに1〜9が揃うように数字を埋めるパズル
- $n \times n$の盤面の一般化数独を考える（普通の数独は$n = 9$）

$O(n)$回のCS/PSS [GNPR07], [SMS18]

- 各マスに答えの数字が3枚ずつ裏向きで置かれている状態を作る
- 検証したい9枚を取り、完全シャッフルし、オープンして確認
- すべての行・列・ブロック（合計27個）について検証を行う

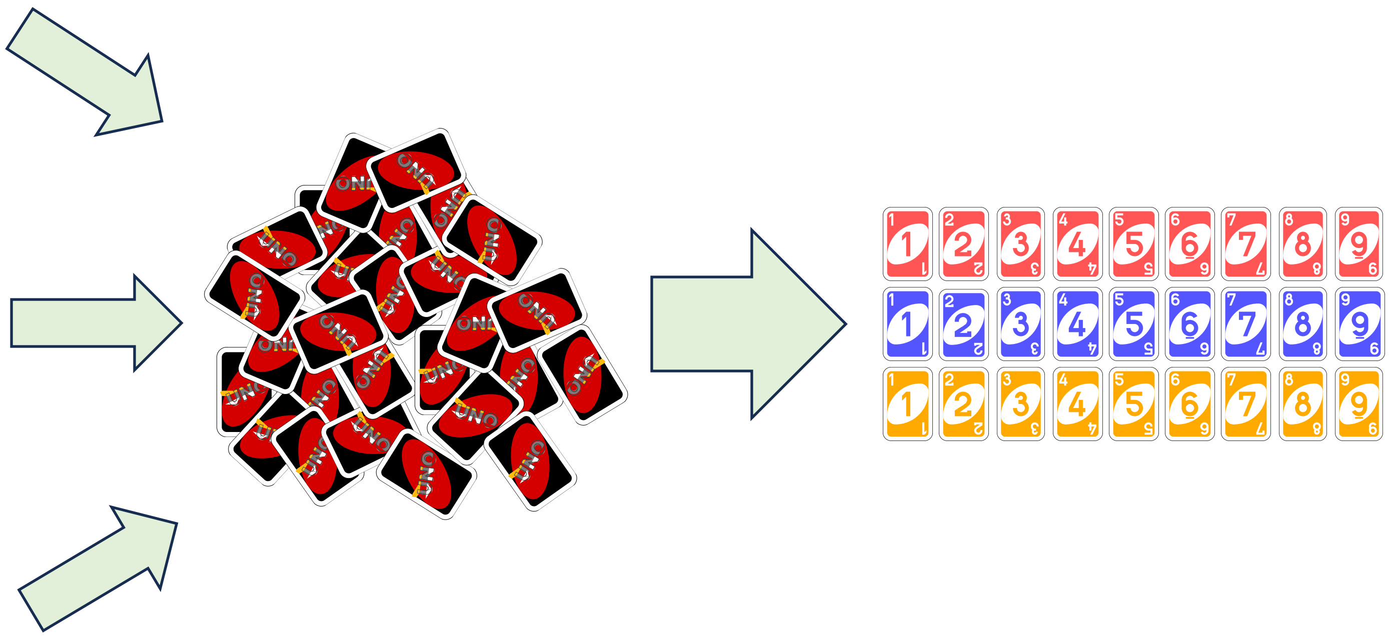


- Gradwohl et al.はこのアイディアで初めての数独ZKPを構成
- Sasaki et al.はパーフェクトな健全性（答えが存在しない数独に対しては検証者は100%の確率で証明を拒否する性質）を達成

[GNPR07] Gradwohl, Naor, Pinkas, Rothblum, Cryptographic and Physical Zero-Knowledge Proof Systems for Solutions of Sudoku Puzzles, FUN 2007.
[SMS18] Sasaki, Mizuki, Sone, Efficient card-based zero-knowledge proof for Sudoku, FUN 2018.

$O(\sqrt{n})$回のPSS [TM23]

- $\sqrt{n}$ 色のカードを用いて、 $\sqrt{n}$個の行検証を同時に行う

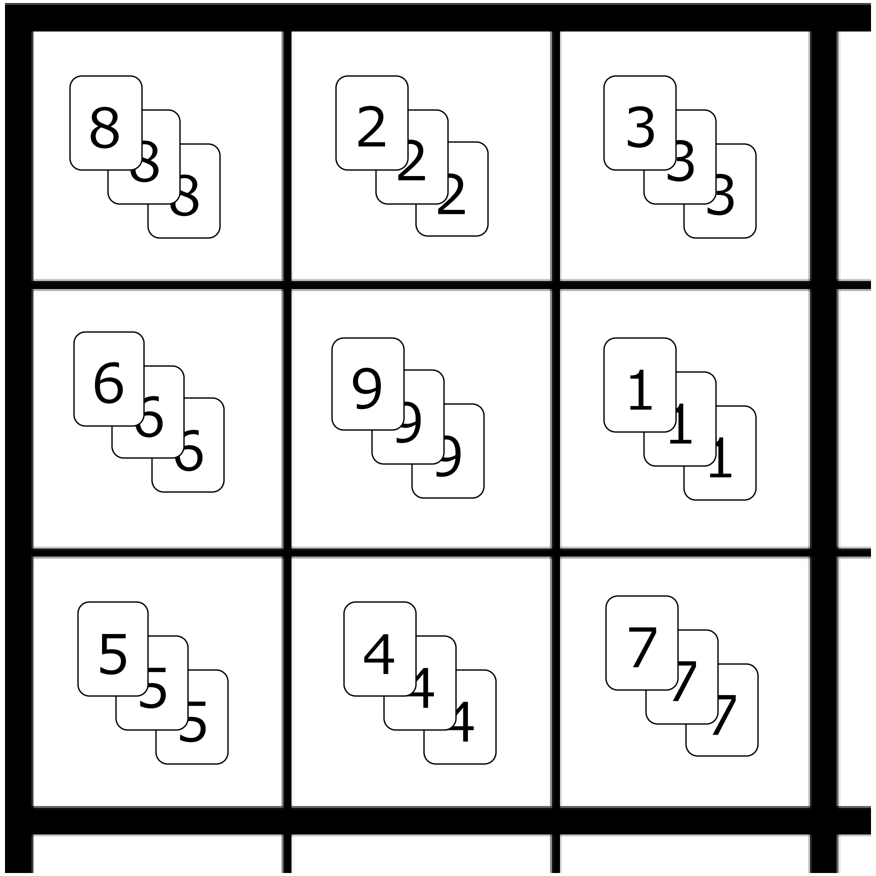


シャッフル回数：$O(n)$回→$O(\sqrt{n})$回

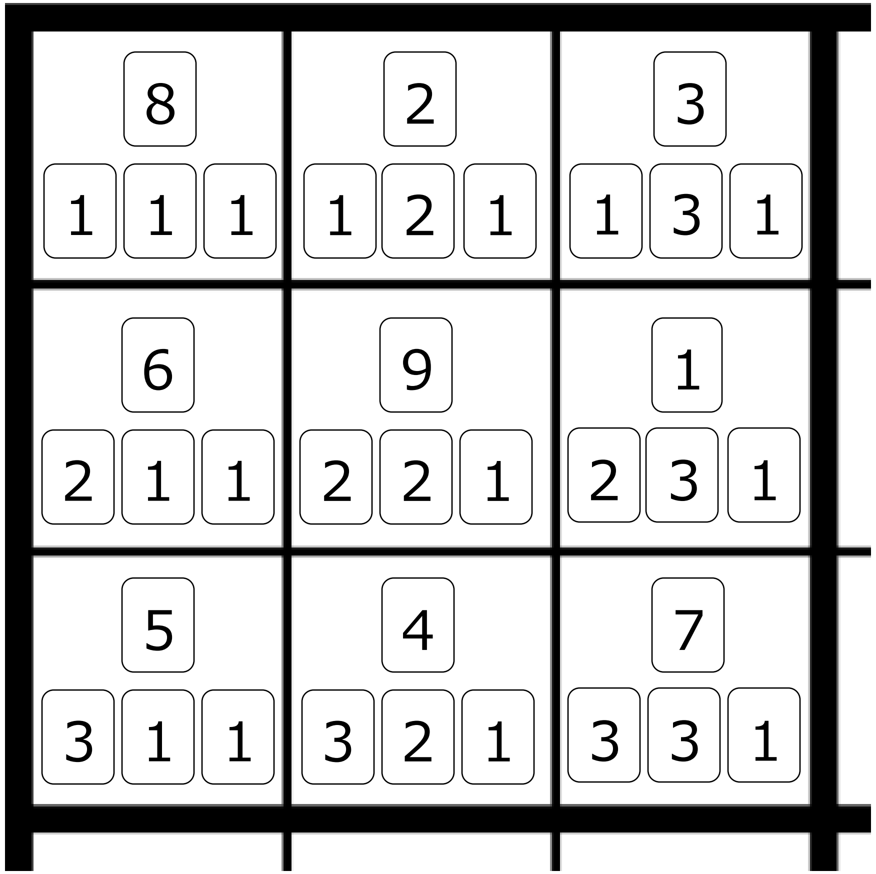
[TM23] Tanaka, Mizuki. Two UNO Decks Efficiently Perform Zero-Knowledge Proof for Sudoku. FCT 2023.

定数回のためのアイディア [TSS+25]

従来：**解答そのもの**を扱う



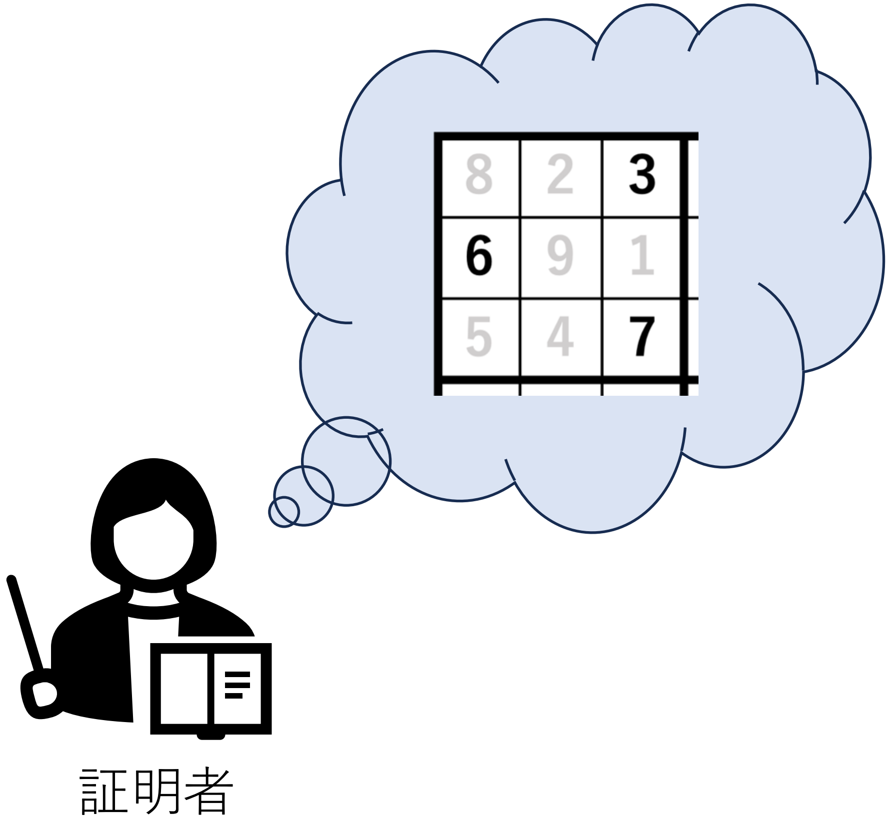
今回：**マスの座標**を扱う

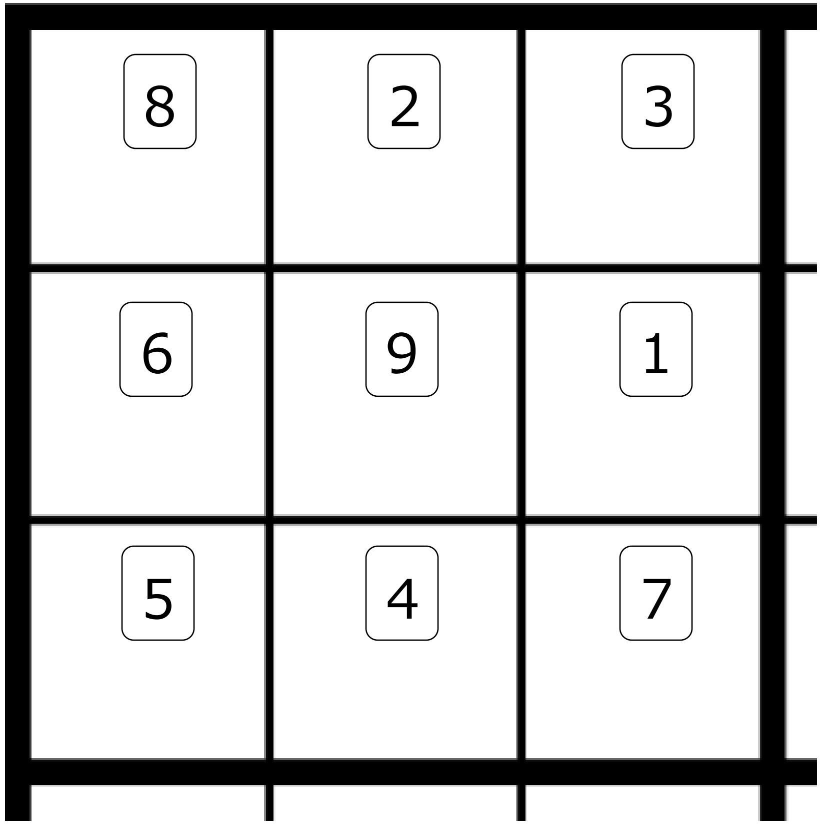


[TSS+25] Tanaka, Sasaki, Shinagawa, Mizuki, Only Two Shuffles Perform Card-Based Zero-Knowledge Proof for Sudoku of Any Size, SOSA 2025.

定数回プロトコル：2回のPSS

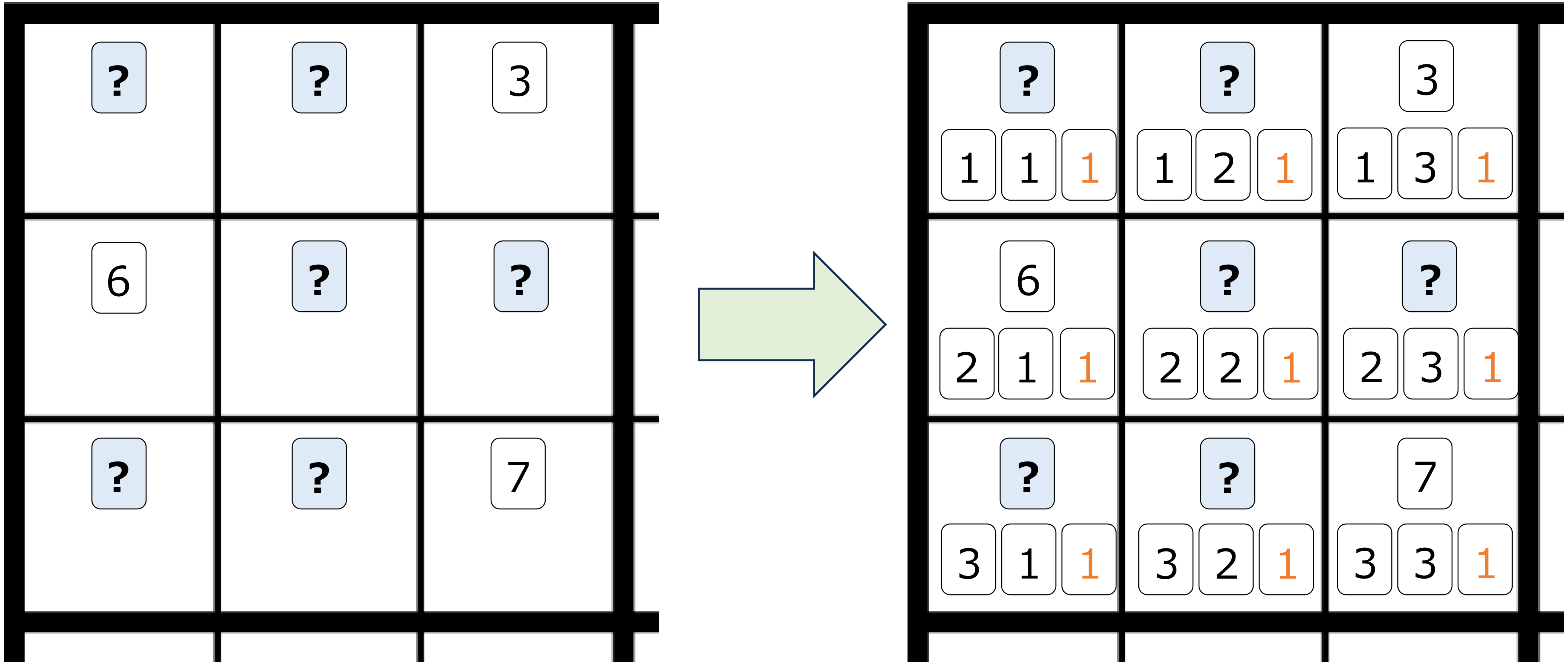
1. 証明者は各マスに対応するカードを**1枚ずつ**裏向きに置く
   - ただし、最初から数字のあるマスは表向きに置く





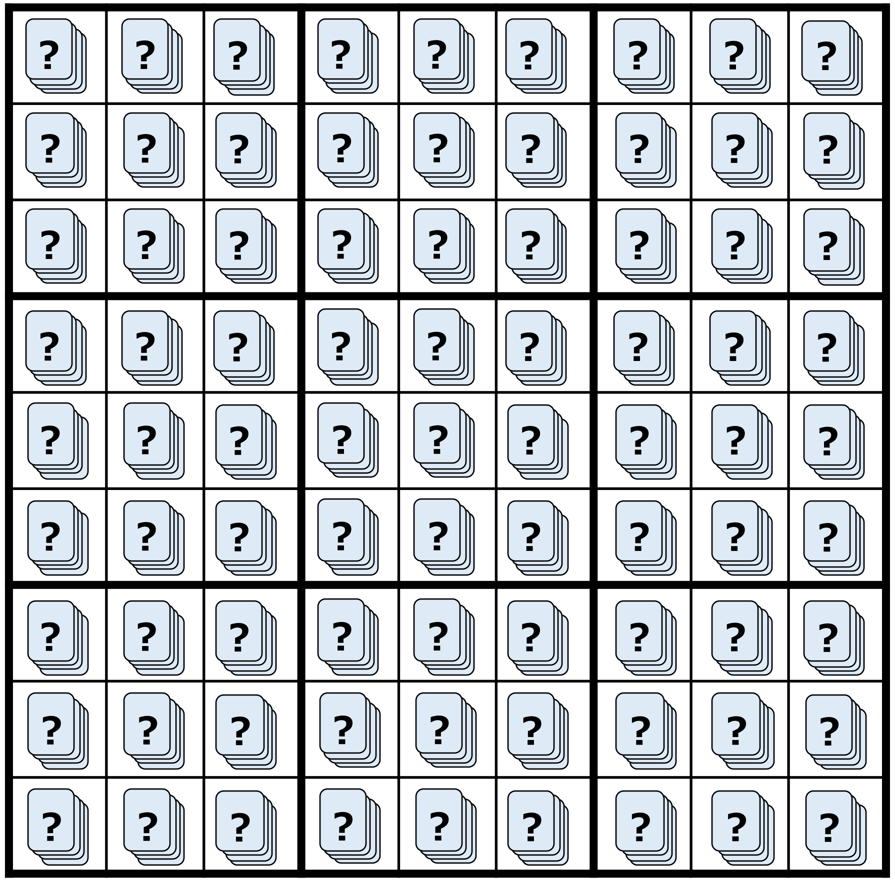
定数回プロトコル：2回のPSS

2. a行b列cブロックのマスに a b c を置く（ブロックだけ赤色）



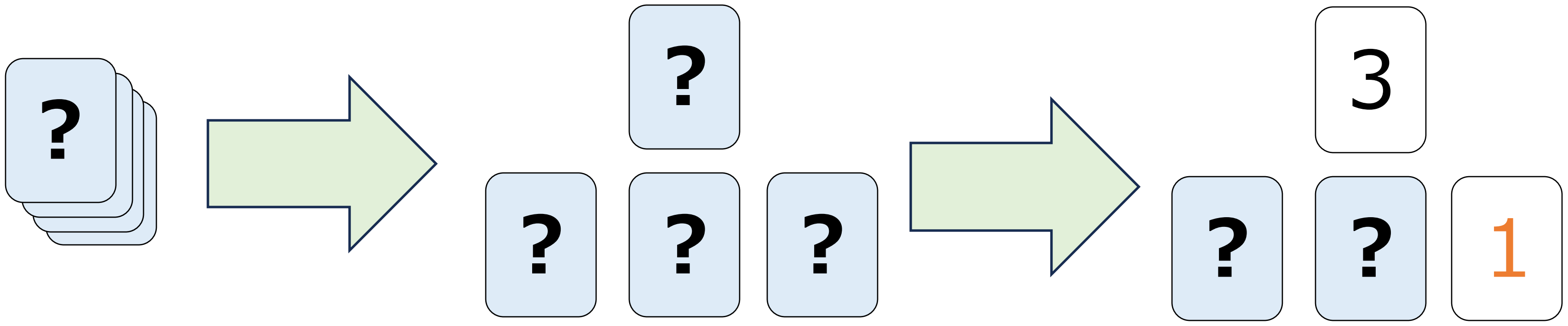
定数回プロトコル：2回のPSS

3. 81個のカード束にパイルスクランブルシャッフルを施す

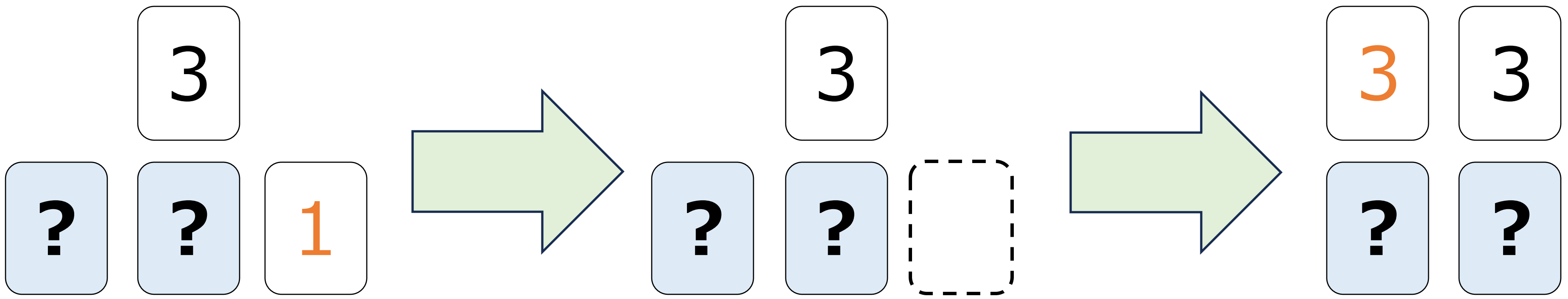


定数回プロトコル：2回のPSS

4.　「解」と「ブロック」をめくり、ブロック検証を行う

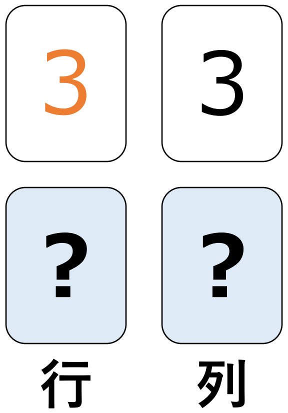


5. ブロック検証後、「解」と同じ数字の赤カードを上に置く



定数回プロトコル：2回のPSS

6. 縦の2枚を束とし、81×2個の束にPSSを施す（2回目のPSS）



7. すべてのカードをオープンし、行検証と列検証を行う

- 「座標」と「2色で行検証と列検証を集約」により2回PSSを達成

対話的設定のプロトコル

- **対話的設定**：プロトコルの操作が証明者の知識（witness）に依存
  - 証明者がwitnessに依存した**秘匿置換**を行う
- **対話的入力**：入力カード列の作成段階のみ対話的設定
  - 検証者がカード確認し、証明者が秘匿置換で入力する

	カード枚数	シャッフル回数	対話的入力	対話的操作
Sasaki et al. [SMM+20]	n^2	$3n+1$	✓	
Ruangwises [Rua21]	$n^2+n(\sqrt{n}+1)+\sqrt{n}$	$4\sqrt{n}$		✓
Ruangwises [Rua21]	$n^2+2n+3\sqrt{n}$	$2n^2(\sqrt{n}-1)+2$		✓
Ono et al. [ORA+24]	$\mathbf{2n^2}$	$\mathbf{1}$	✓	

[SMM+20] T. Sasaki, D. Miyahara, T. Mizuki, H. Sone, Efficient card-based zero-knowledge proof for Sudoku, TCS 2020.
[Rua21] S. Ruangwises, Two Standard Decks of Playing Cards Are Sufficient for a ZKP for Sudoku, COCOON 2021.
[ORA+25] T. Ono, S. Ruangwises, Y. Abe, K. Hatsugai, and M. Iwamoto, Single-Shuffle Physical Zero-Knowledge Proof for Sudoku Using Interactive Inputs, APKC 2025.

ZKPに関する未解決問題

- 未解決問題：PSS 1回の(非対話の)プロトコルは構成可能か？
  - 各マスに1枚ずつ**数字カード**を置く設定
- 未解決問題：PSS 1回の(対話型の)プロトコルの他の構成は？
  - カード枚数、秘匿置換回数、シャッフル種類、操作性の高さ
- 未解決問題：9×9数独ZKPを5分以内で実行可能な実装方法は？
- 他のパズルに対する定数回のプロトコル
  - グラフ同型[MHM21]、巡回セールスマン[猪狩ら24]

[MHM21] Miyahara, Haneda, Mizuki. Card-based Zero-knowledge Proof Protocols for Graph Problems and Their Computational Model. ProvSec 2021.
[猪狩ら24] 猪狩, 駒野, 水木. 巡回セールスマン問題の物理的ゼロ知識証明プロトコル. DICOMO 2024.

まとめ

- MPCのシャッフル回数に関する未解決問題
  - PSS 1回のプロトコルを構成できるか？
  - PSS 2回・一様閉 1回・一様 1回の場合のカード枚数を削減できるか？
  - 他のシャッフル（例えば巡回群シャッフル）の最小回数は？

- ZKPのシャッフル回数に関する未解決問題
  - PSS 1回の(非対話の)数独ZKPは構成可能か？
  - PSS 1回の(対話型の)数独ZKPの他の方式は？
  - 9×9の数独ZKPを5分以内で実行できるか？
  - 他のパズルに対する定数回のプロトコル
    - 数独・グラフ同型問題・巡回セールスマン問題