

連動操作を用いる上下カード n 枚しきい値関数プロトコルの 不可能性証明

飯野 静流

本発表は次の論文に基づく：

Shizuru Iino, Yang Li, Kazuo Sakiyama, Daiki Miyahara,
“An n -Card Threshold Protocol Is Impossible”, CCWC2026, (Jan., 2026)

1. 背景
 - i. カードベース暗号
 - ii. 上下カード版Five-Card Trick
 - iii. 貢献
2. 準備
 - i. しきい値関数
 - ii. 定式化モデル
3. 不可能性証明
4. $n + 1$ 枚 n 入力コミット型閾値プロトコル
5. 結論

- カードベース暗号は物理的なカード組を用いて秘密計算を達成する暗号方式
 - 入力情報の一部も漏洩しないまま結果だけを得る
- 上下カード組で $x \in \{0,1\}$ を表現する方法:



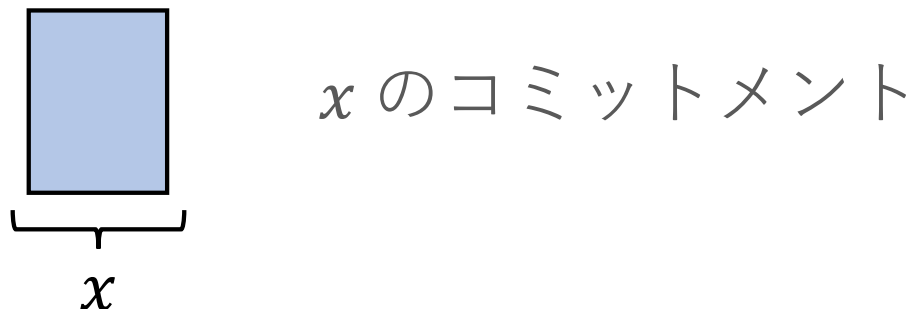
→ 裏面（上下対称）にした1枚の呼称:



- カードベース暗号は物理的なカード組を用いて秘密計算を達成する暗号方式
 - 入力情報の一部も漏洩しないまま結果だけを得る
- 上下カード組で $x \in \{0,1\}$ を表現する方法：

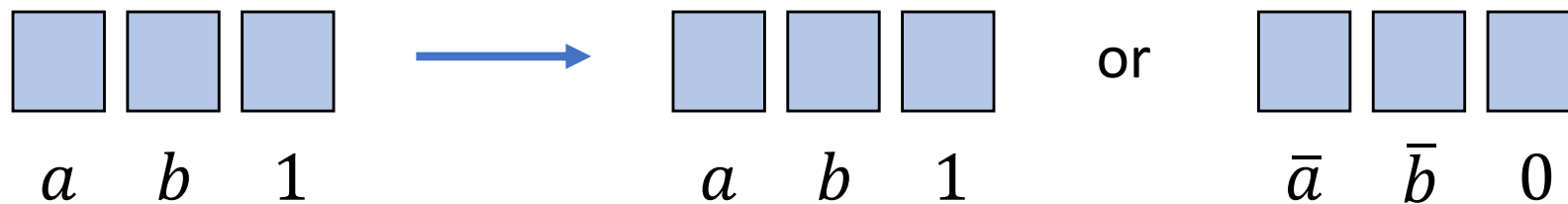


→ 裏面（上下対称）にした1枚の呼称：

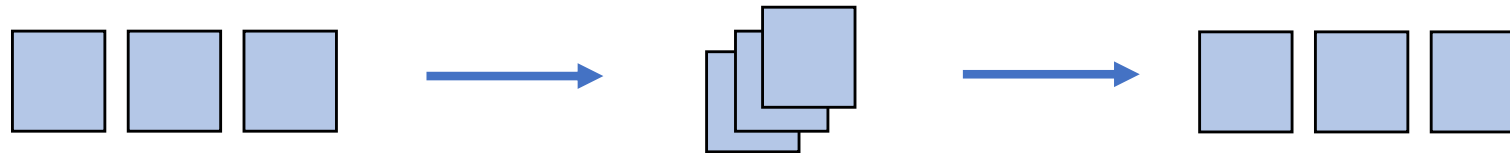


入力方式 $\downarrow$ $x = 0$ | $\uparrow$ $x = 1$ 裏: 

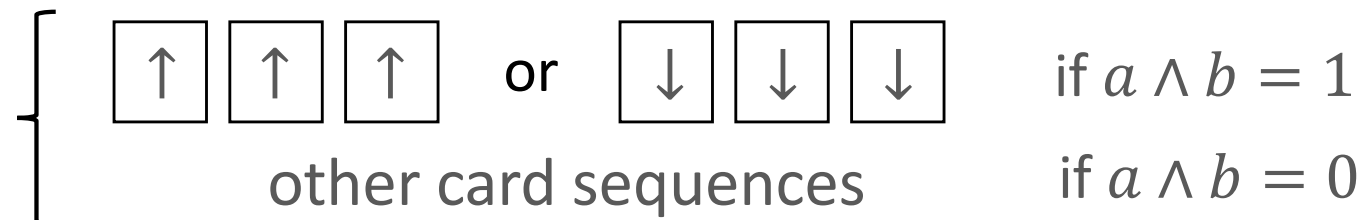
1. 2入力 + 追加カード1枚を連動して回転させる:



2. 完全ランダムに並び替える:



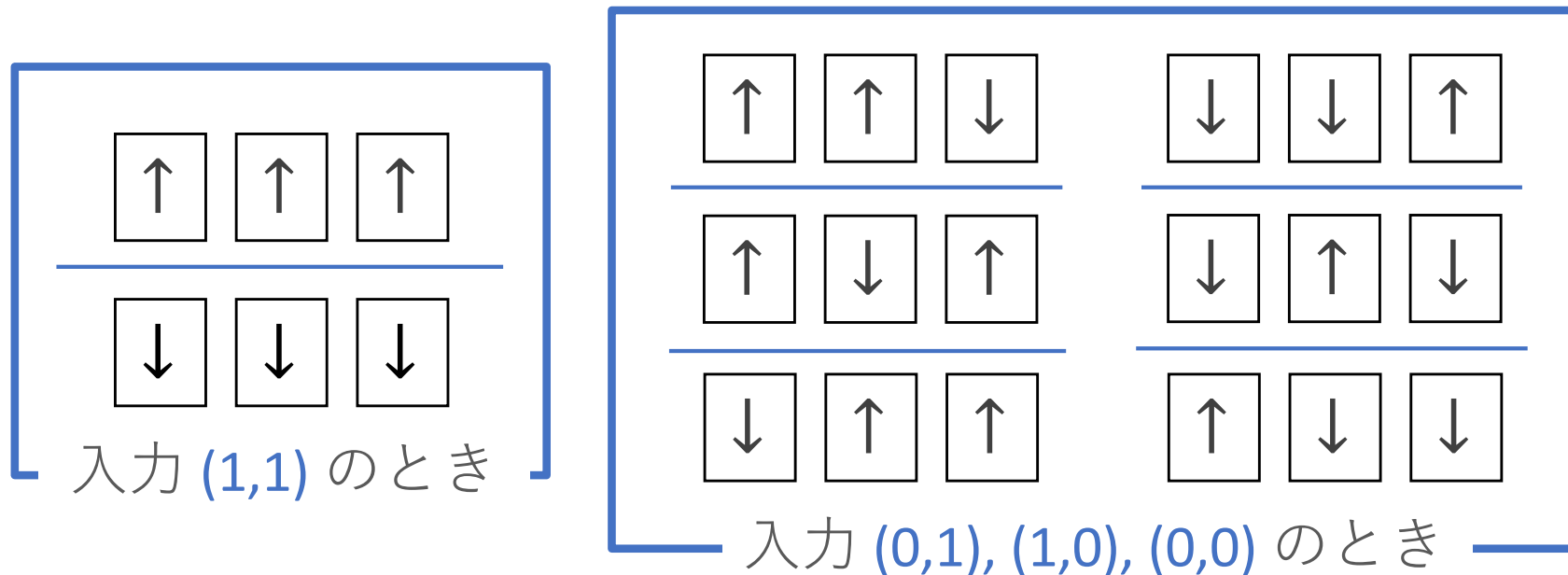
3. 全てめくる:



正当性と 安全性

- 正当性: $a \wedge b$ の計算を正しく行う

操作3のカード列の可能性



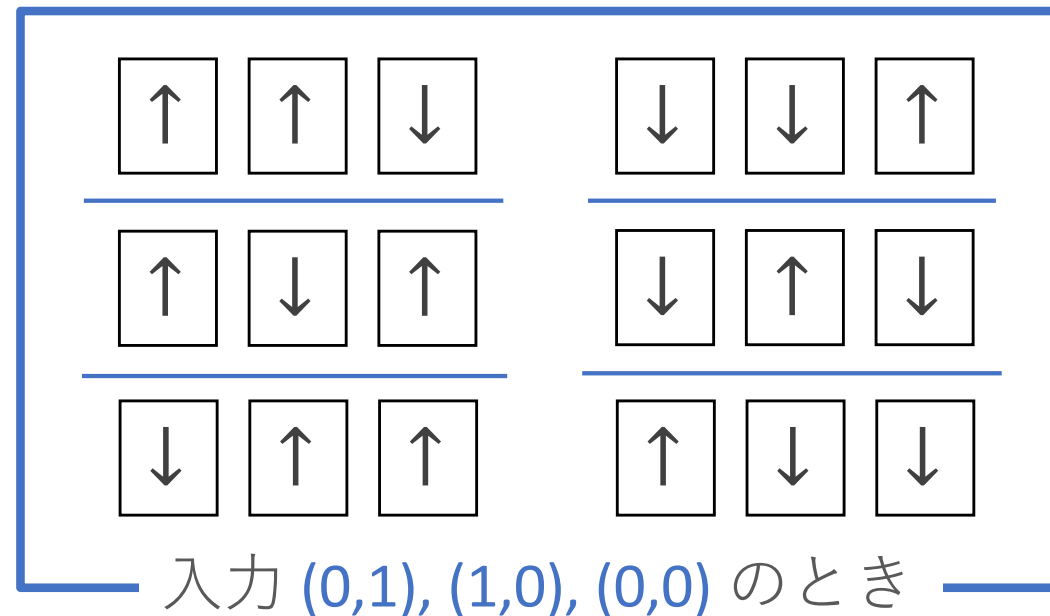
この2つのグループが共通の要素を持たない

= 正当性を満たす

正当性 と 安全性

- 安全性: 結果から入力値が一切漏れない

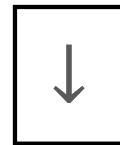
操作3のカード列の可能性



特定の入力を識別できる一意なカード列は存在しない

= 安全性を満たす

入力方式



$x = 0$

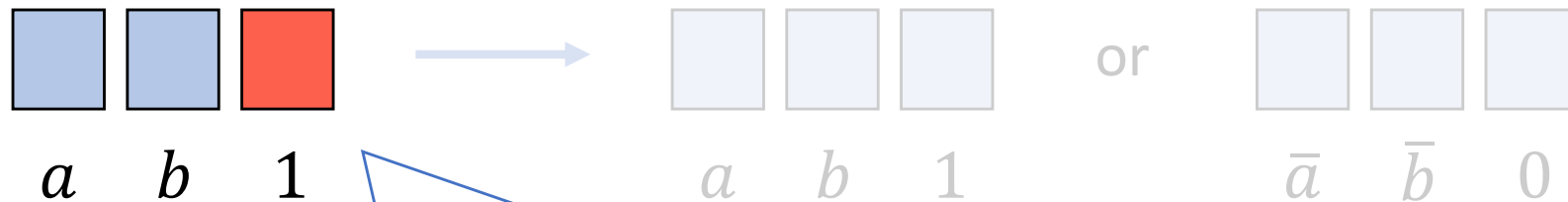


$x = 1$

裏:



1. 2入力 + 追加カード1枚を連動して回転させる:

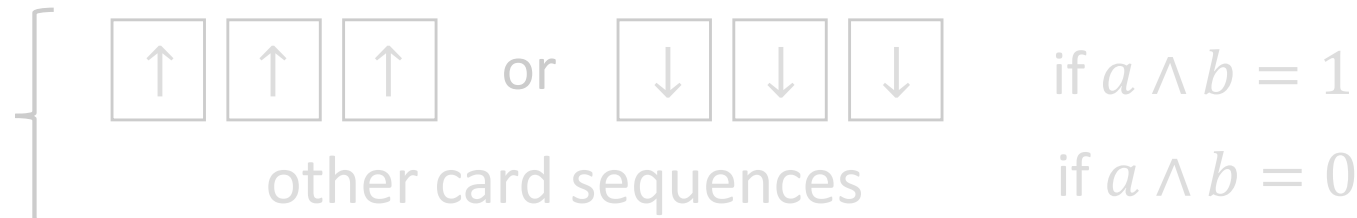


2. 完全ランダムに替える:



この追加カード1枚が必須なのかが未解決

3. 全てめくる:



未解決問題：AND関数

- 設定：上下カードの有限時間コミット型プロトコル
- n 入力ANDは $(n+1)$ 枚で計算できる
 - 構成法：3枚ANDプロトコル[MS14]の繰り返し
- **n 入力ANDは n 枚では計算できないことを示せ**
 - $n=2$ のときは解決済み
 - 二色カードの4枚ANDが不可能であること[KWH15]からただちに従う
 - n が3以上のときは未解決

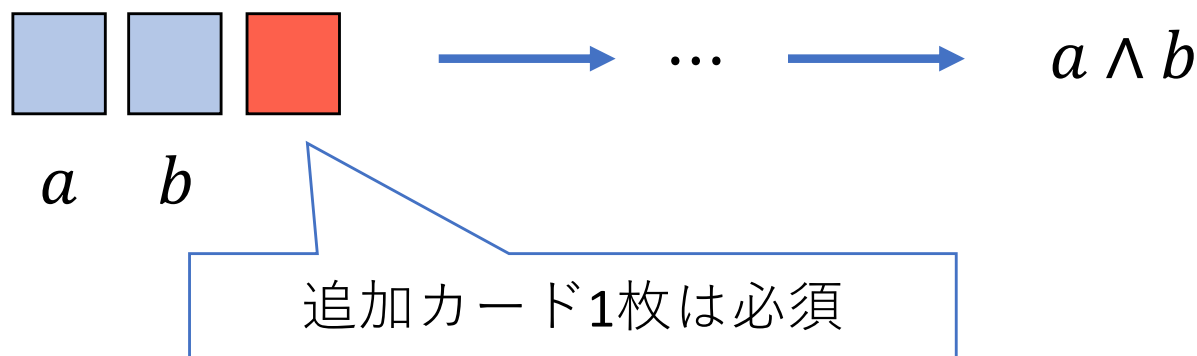
[MS14] Takaaki Mizuki, Hiroki Shizuya. Practical Card-Based Cryptography. FUN 2014.

[KWH15] Alexander Koch, Stefan Walzer, Kevin Härtel. Card-Based Cryptographic Protocols Using a Minimal Number of Cards. ASIACRYPT 2015. 8

引用：品川和雅, “カードベース暗号に登場するさまざまなカード組と符号化,” 産学連携によるカードベース暗号の数理的未解決問題と新課題の整理 | 2023a020

不可能性証明

- 2枚2入力ANDプロトコルは存在しない
 - 有限時間やコミット型に限らない



- この結果を3入力以上の場合にも適用し、 n 枚 n 入力しきい値プロトコルの不可能性に拡大

構築

- $n + 1$ 枚 n 入力コミット型しきい値プロトコル

1. 背景
 - i. カードベース暗号
 - ii. 上下カード版Five-Card Trick
 - iii. 貢献
2. 準備
 - i. しきい値関数
 - ii. 定式化モデル
3. 不可能性証明
4. $n + 1$ 枚 n 入力コミット型閾値プロトコル
5. 結論

- n 入力のしきい値関数
定義

$$\text{TH}_n^t(x_1, x_2, \dots, x_n) = \begin{cases} 0 & \text{if } \sum_{i=1}^n x_i \leq t \\ 1 & \text{otherwise} \end{cases}$$

- 論理積は $\text{AND} = \text{TH}_n^{n-1}(x_1, x_2, \dots, x_n)$
- 多数決は $\text{maj}(x_1, x_2, \dots, x_n) = \text{TH}_n^{\lfloor n/2 \rfloor}(x_1, x_2, \dots, x_n)$
- n 入力しきい値関数の不可能性を示せば、
論理積と多数決などの特別なケースにおいても不可能性が示される

Mizuki–Shizuya Model^[5], Shinagawa and Nuida’s extension^[6]

- turn: カードをめくる操作

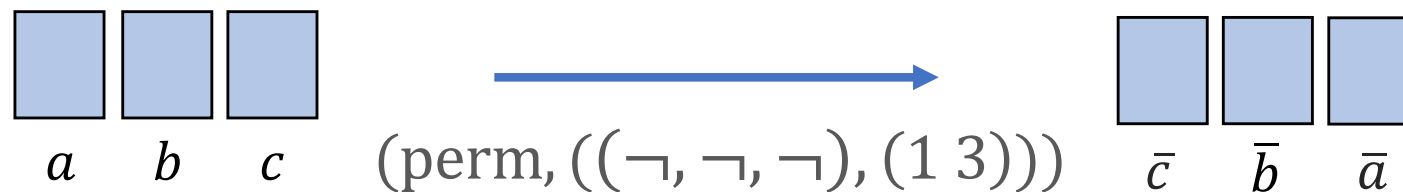
例



- **extended perm**: 置換 π に従い、カードを並べ替える操作

例 $\pi = ((\neg, \neg, \neg), (1\ 3))$

上下の向きを変える反転操作と巡回置換を同時に行う（連動操作）



[5] T. Mizuki and H. Shizuya, “A formalization of card-based cryptographic protocols via abstract machine,” Int. J. Inf. Secur., 2014.

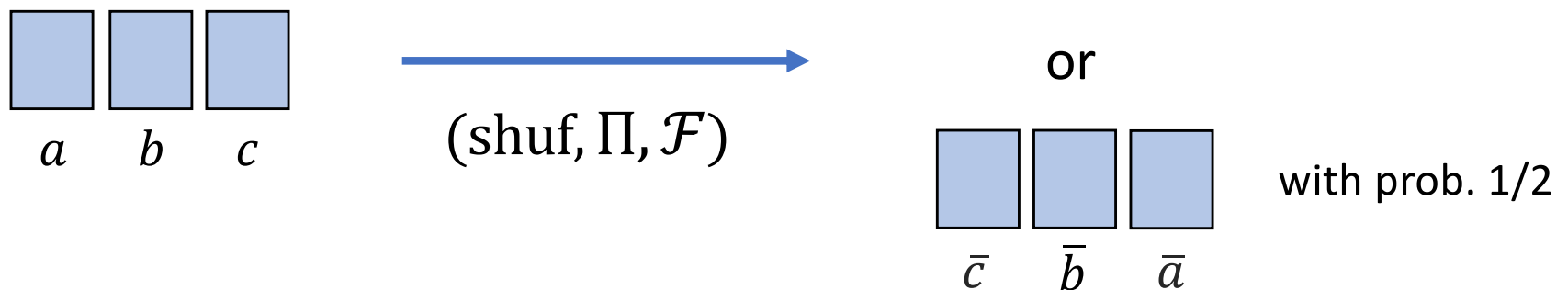
[6] K. Shinagawa and K. Nuida, “Card-based protocols imply PSM protocols,” STACS 2025

Mizuki–Shizuya Model^[5], Shinagawa and Nuida’s extension^[6]

- shuf: シャッフル操作
確率分布 $\mathcal{F}$ に従い、置換集合 Π の要素 π を選択し、
その置換を適用する

例

- $\Pi = \{((\text{id}, \text{id}, \text{id}), \text{id}), ((\neg, \neg, \neg), (1\ 3))\}$
- $\mathcal{F}$ は一様分布
(=ランダムに選択される)



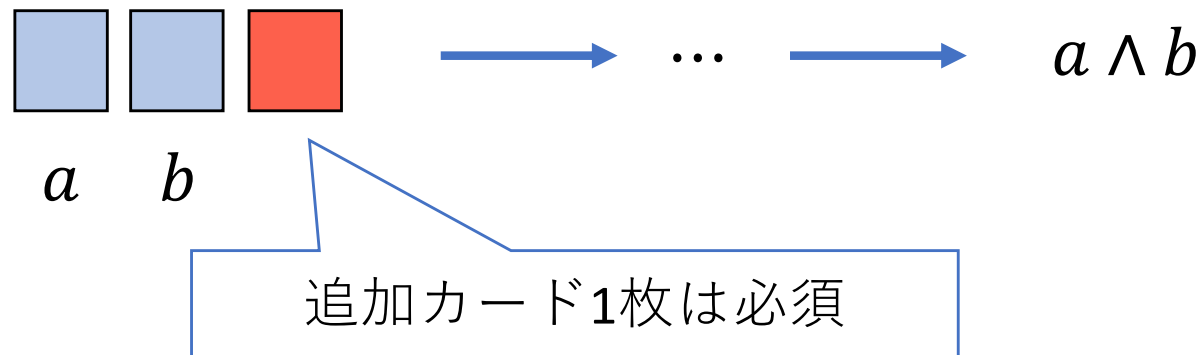
[5] T. Mizuki and H. Shizuya, "A formalization of card-based cryptographic protocols via abstract machine," Int. J. Inf. Secur., 2014.

[6] K. Shinagawa and K. Nuida, "Card-based protocols imply PSM protocols," STACS 2025

1. 背景
 - i. カードベース暗号
 - ii. 上下カード版Five-Card Trick
 - iii. 貢献
2. 準備
 - i. しきい値関数
 - ii. 定式化モデル
3. 不可能性証明
4. $n + 1$ 枚 n 入力コミット型閾値プロトコル
5. 結論

不可能性証明

- 2枚2入力ANDプロトコルは存在しない
 - 有限時間やコミット型に限らない



- この結果を3入力以上の場合にも適用し、 n 枚 n 入力しきい値プロトコルの不可能性に拡大

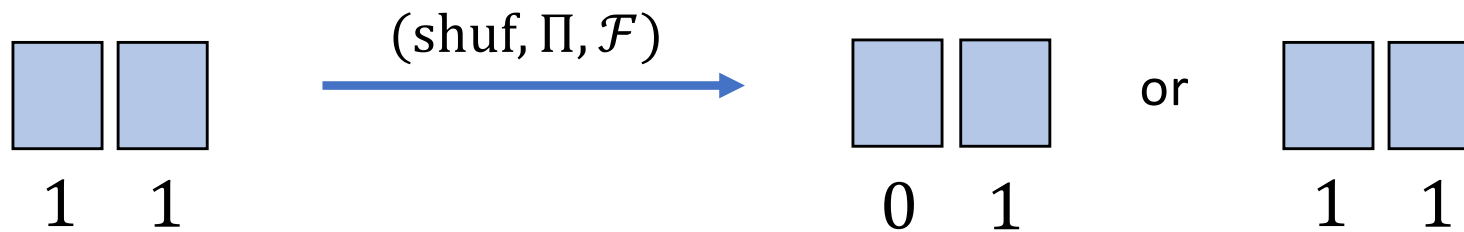
構築

- $n + 1$ 枚 n 入力コミット型しきい値プロトコル

アイデア: (1,1)が他と必ず衝突する

直感的なアイデア

- 回転を含むシャッフルの場合:



- $\Pi = \{((\neg, \text{id}), \text{id}), \text{id}\}$
(= 1枚目を回転)

正当性の違反

入力 (1, 1) の操作後の
カード列が入力 (0, 1) と同一に

- 回転を含まないシャッフルの場合:



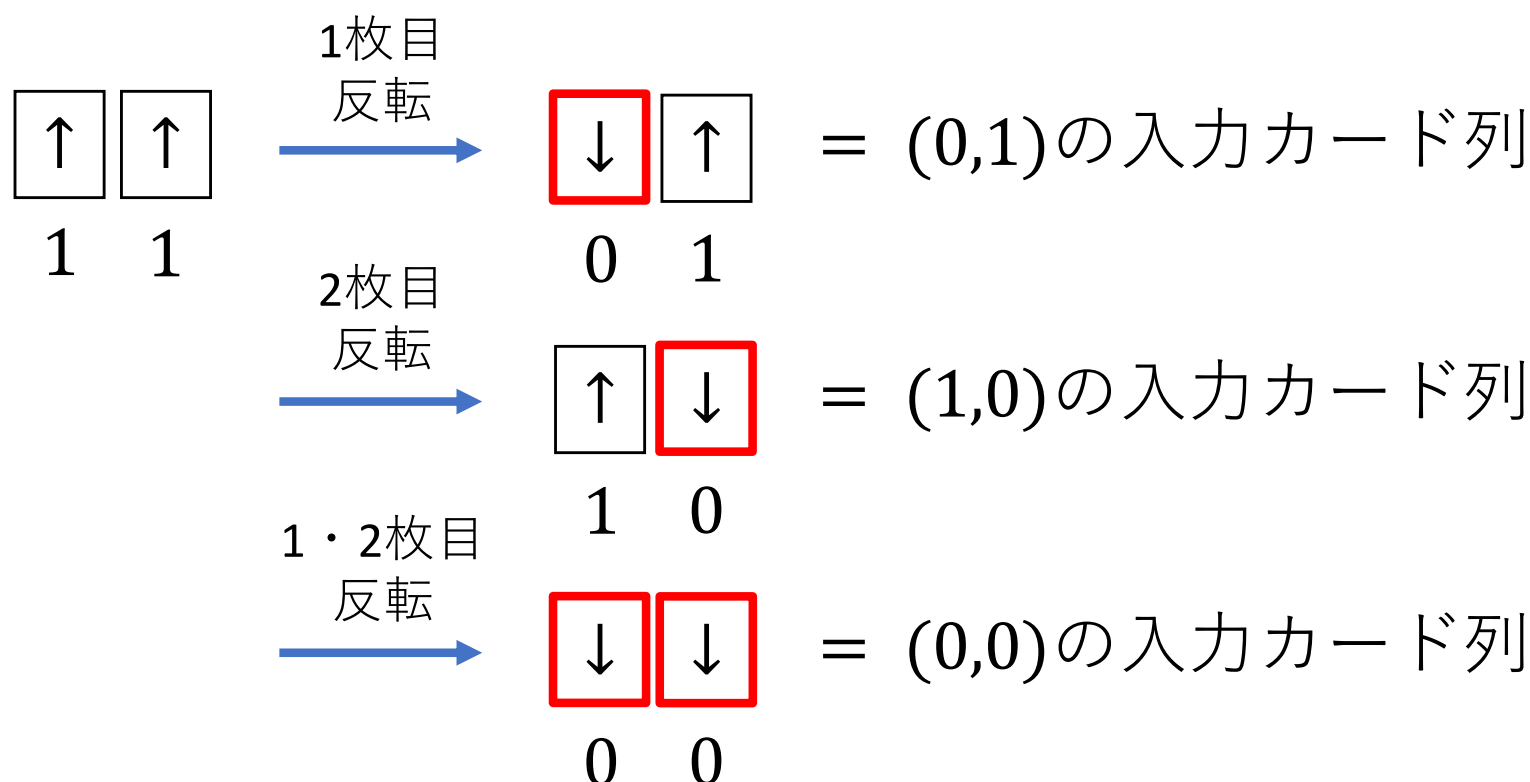
安全性の違反

結果のカード列が
区別可能に

id : 恒等 (拡張) 置換

回転を含むシャッフルのとき

例 $(\text{shuf}, \{\mathbf{id}, \pi\}, \mathcal{F})$ の適用後 (**id** 固定・省略)

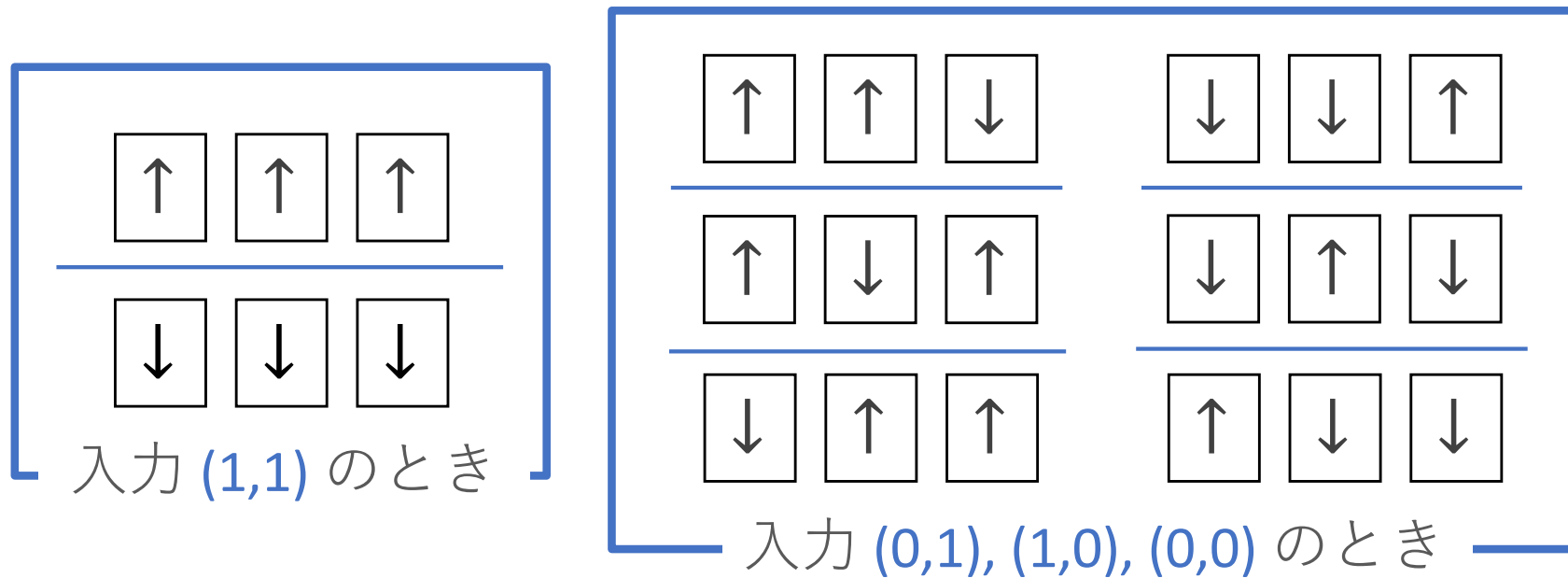


- $\pi(1, 1) \in \{(0, 0), (0, 1), (1, 0)\}$ if π includes negations

正当性 と 安全性

- 正当性: $a \wedge b$ の計算を正しく行う
- 安全性: 結果から入力の値が一切漏れない

操作3のカード列の可能性

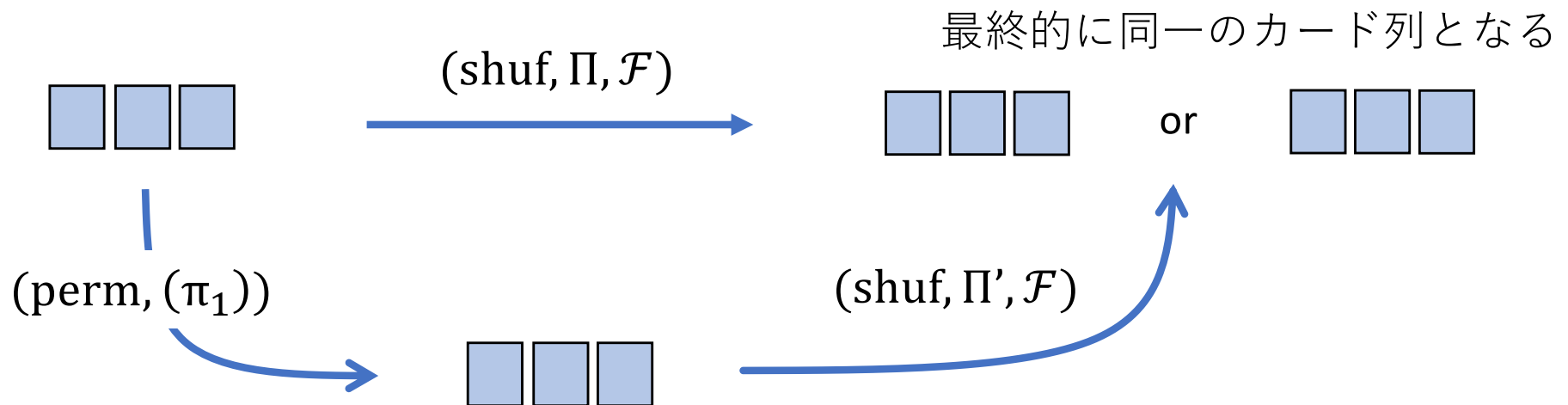


- シャッフルの置換集合は id を含む形に変更できるため、id を仮定しても一般性は保たれる

例 拡張置換集合 $\Pi = \{\pi_1, \pi_2\}$ を考える

集合の一つの元 π_1 を基準に取り、
各元に π_1^{-1} を掛けると

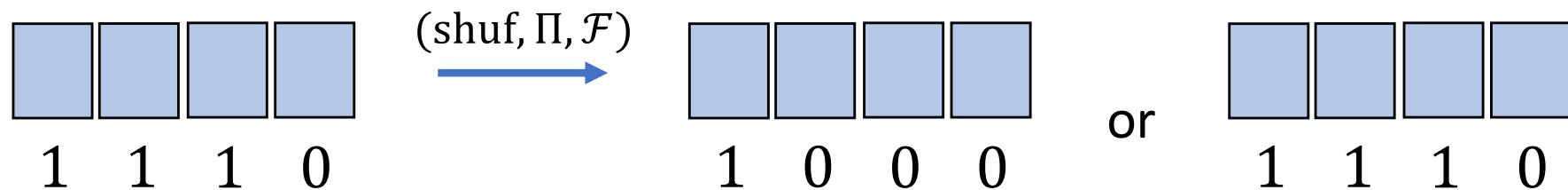
$$\begin{aligned}\Pi' &= \{\pi_1^{-1}(\pi_1), \pi_1^{-1}(\pi_2)\} \\ &= \{\text{id}, \pi_2 \cdot \pi_1^{-1}\}\end{aligned}$$



不可能性証明: n 入力しきい値関数

直感的なアイデア (n 入力、しきい値 t)

- 回転を含むシャッフルの場合:



$\neg$: 上下反転記号
id: 恒等 (拡張) 置換

- $\Pi = \{((\text{id}, \neg, \neg, \text{id}), \text{id}), \text{id}\}$
($= 2 \cdot 3$ 枚目が回転する)

正当性の違反

- どんな回転を含むシャッフルでも
 t 付近の入力で重なることがある

しきい値 t 以下の
入力カード列と同一に

- 回転を含まない
シャッフルの場合:

安全性の違反

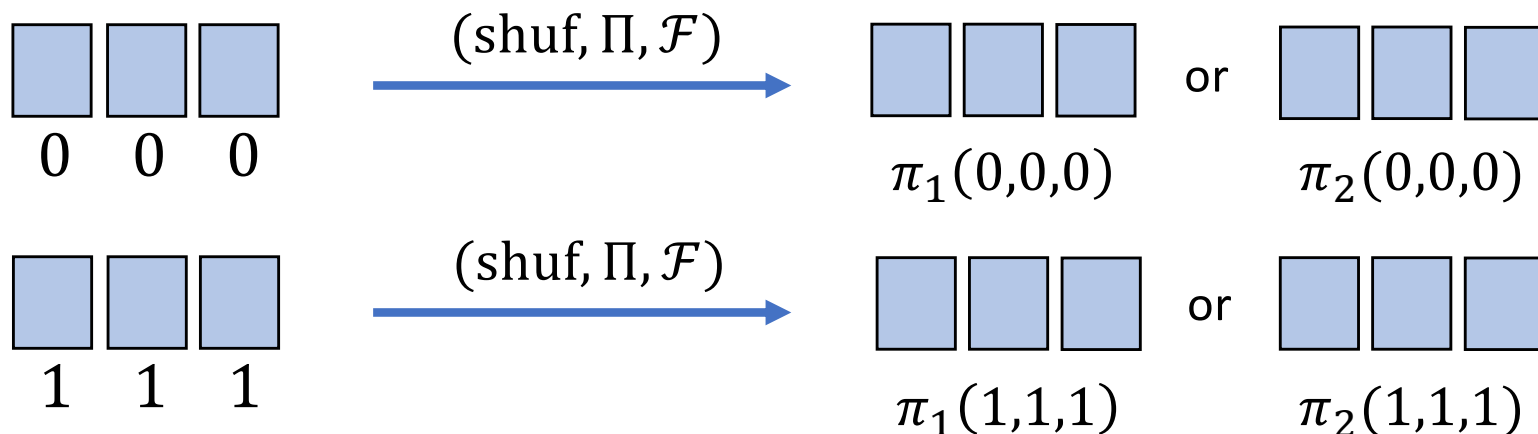
結果のカード列が区別可能に

補題1.

正当性を仮定すると、
出力の異なる2つの入力に対応するカード列はどのような
シャッフル操作によっても**混ざることはない**

例 $\Pi = \{\pi_1, \pi_2\}$ のとき

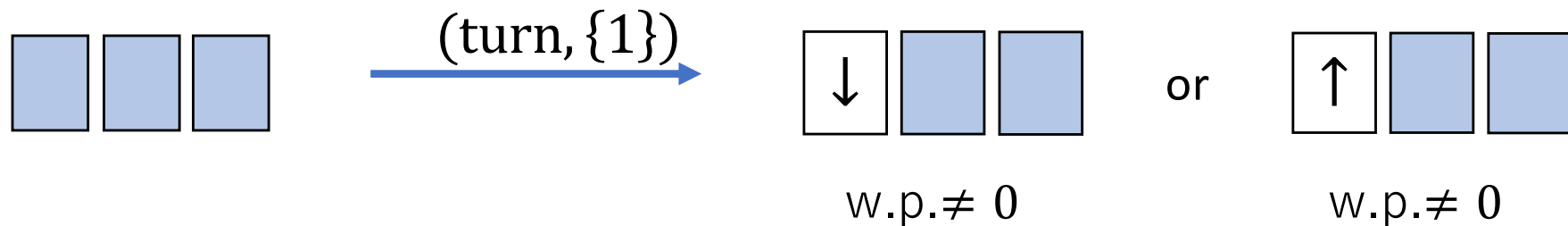
- $\pi_1(0,0,0) \neq \pi_1(1,1,1)$ かつ $\pi_1(0,0,0) \neq \pi_2(1,1,1)$
- $\pi_2(0,0,0) \neq \pi_1(1,1,1)$ かつ $\pi_2(0,0,0) \neq \pi_2(1,1,1)$



補題2.

安全性を仮定すると、
初めにシャッフル操作を行い、その直後に **turn** 操作を行う際
常に**0**もしくは**1**が確率的に現れる

例



しきい値プロトコルの不可能性

証明.

補題2より shuf 操作前に turn 操作はできない

(shuf, Π , $\mathcal{F}$) の適用後を検討、 $\Pi \supseteq \{\text{id}, \pi\}$ を仮定

- 入力の合計値 s がしきい値 t 付近の入力に注目

入力例	出力 ($t = 2$)
0000	0
0001	0
0011	0
1100	0
1110	1
0111	1
1111	1

上下反転したとき
 s の値が変動する

$s = 2$

$s = 3$

k 枚反転するなら
変化量は 0 から k

s が t を跨いで変化

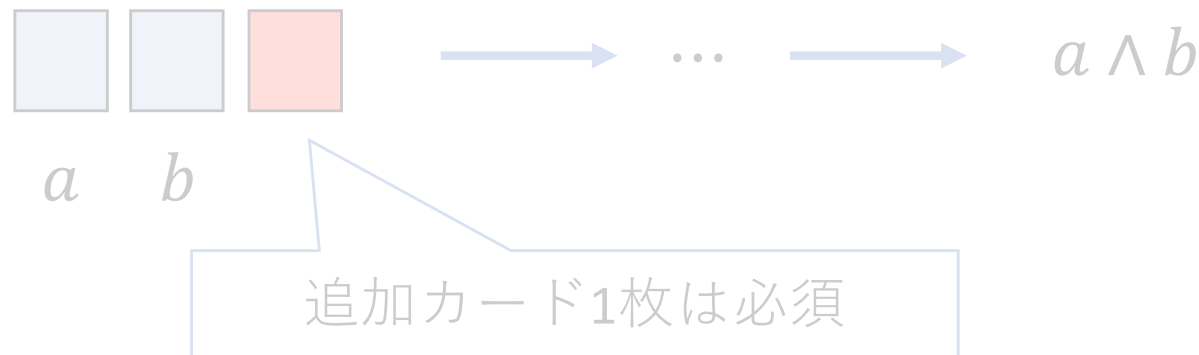


異なる出力グループの
入力が重なる
(補題1に反する)

1. 背景
 - i. カードベース暗号
 - ii. 上下カード版Five-Card Trick
 - iii. 貢献
2. 準備
 - i. しきい値関数
 - ii. 定式化モデル
3. 不可能性証明
4. $n + 1$ 枚 n 入力コミット型閾値プロトコル
5. 結論

不可能性証明

- 2枚2入力ANDプロトコルは存在しない
 - 有限時間やコミット型に限らない



- この結果を3入力以上の場合にも適用し、 n 枚 n 入力しきい値プロトコルの不可能性に拡大

構築

- $n + 1$ 枚 n 入力コミット型しきい値プロトコル

n 入力しきい値プロトコル

- n 入力のしきい値関数
定義

$$\text{TH}_n^t(x_1, x_2, \dots, x_n) = \begin{cases} 0 & \text{if } \sum_{i=1}^n x_i \leq t \\ 1 & \text{otherwise} \end{cases}$$

- プロトコルの仕組み
例 ($n = 8, t = 3$ のとき)

入力 (0, 1, 0, 1, 1, 1, 1, 0)

ソート後 (1, 1, 1, **1**, 1, 0, 0, 0)

$t + 1$ 枚目を開示

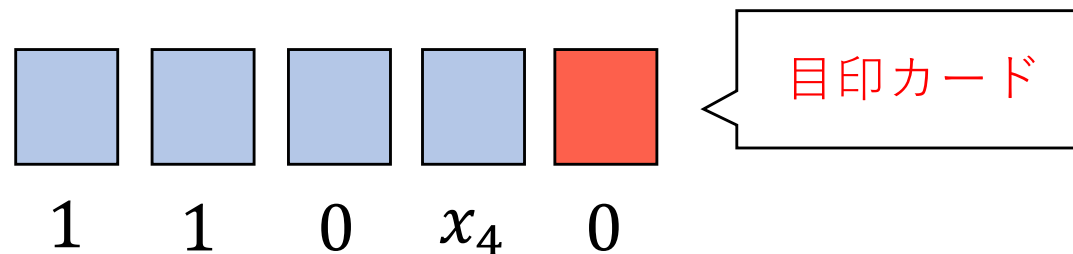
降順ソート

Takahashiらのランダム
反転プロトコル^[9]
を上下カード用に
変換

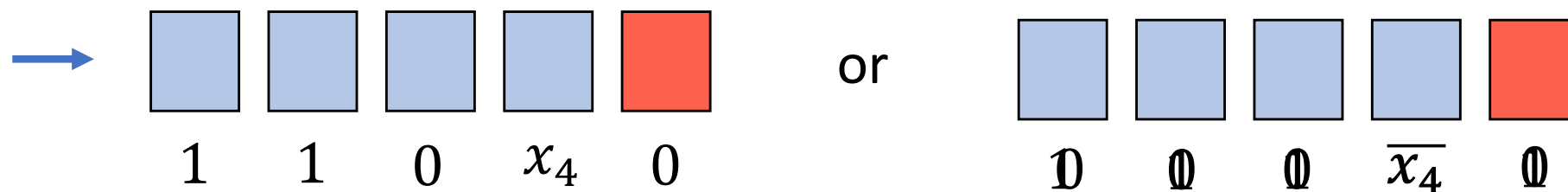
[9] Y. Takahashi, K. Shinagawa, H. Shikata, and T. Mizuki. "Efficient card-based protocols for symmetric functions using four-colored decks." In APKC 2024. ACM, 2024.

n 入力しきい値プロトコル

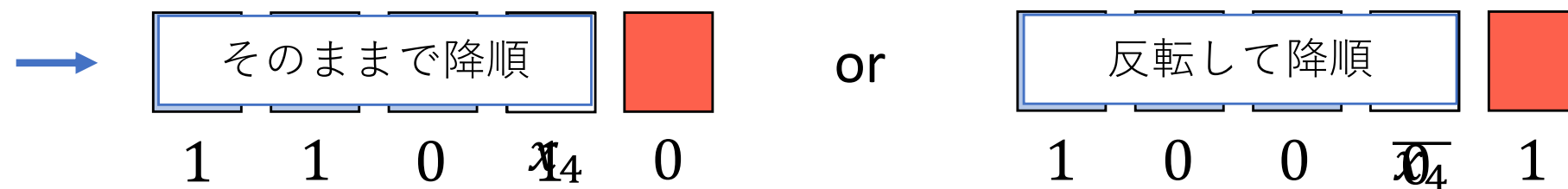
動作例：降順にソートされたカード列 $(1, 1, 0)$ に $x_4 (= 1)$ を追加する



step a : (shuf, Π , $\mathcal{F}$) where $\Pi = \{\text{id}, ((\neg, \neg, \neg, \neg, \neg), (1\ 3))\}$



step b : (perm, ((id, id, id, id, id), (1\ 2\ 3\ 4))) if (turn, {4}) = {↑}

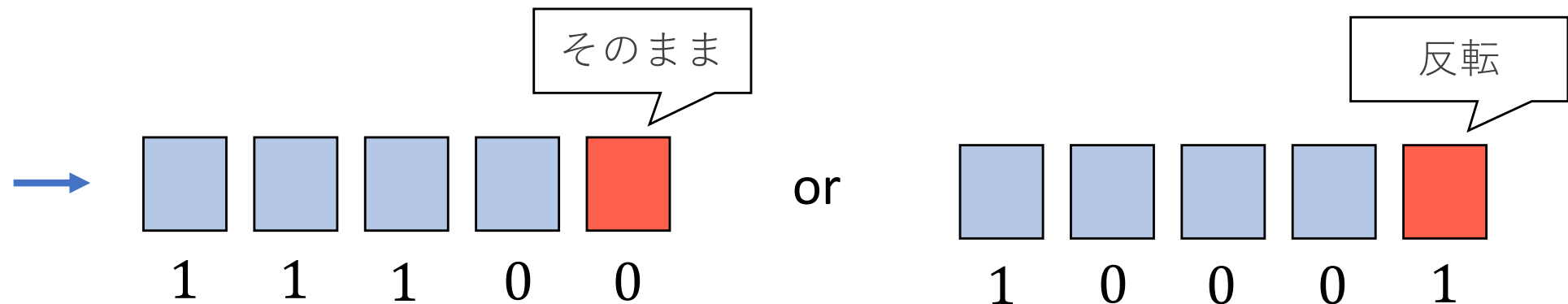


0か1がランダムに現れるので
めくっても安全

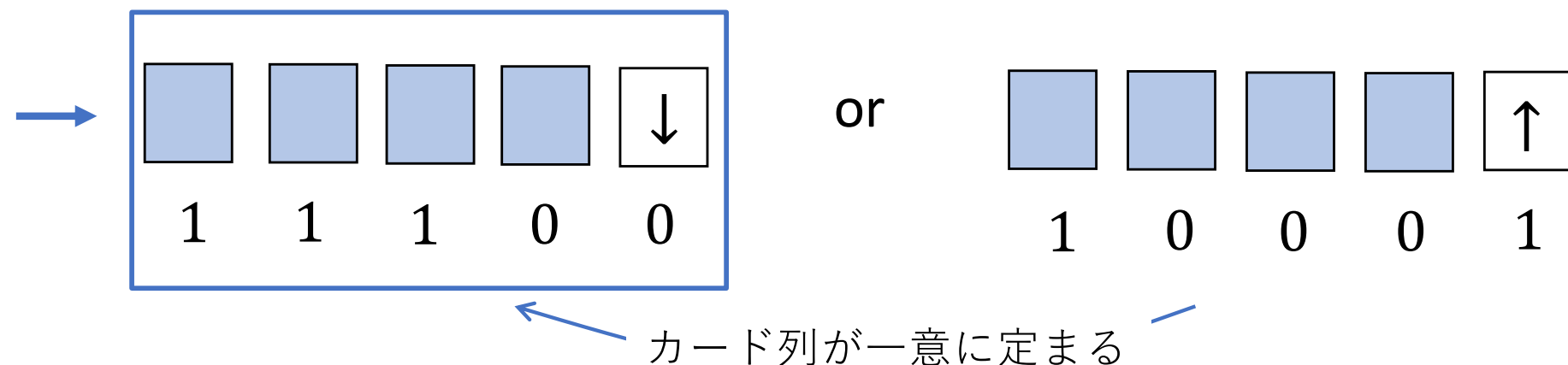
n 入力しきい値プロトコル

step b後は4枚目の位置がわかり、目印カードをめくれない

step c : (shuf, Π , $\mathcal{F}$) where $\Pi = \{\mathbf{id}, ((\neg, \neg, \neg, \neg, \neg), (1\ 4)(2\ 3))\}$



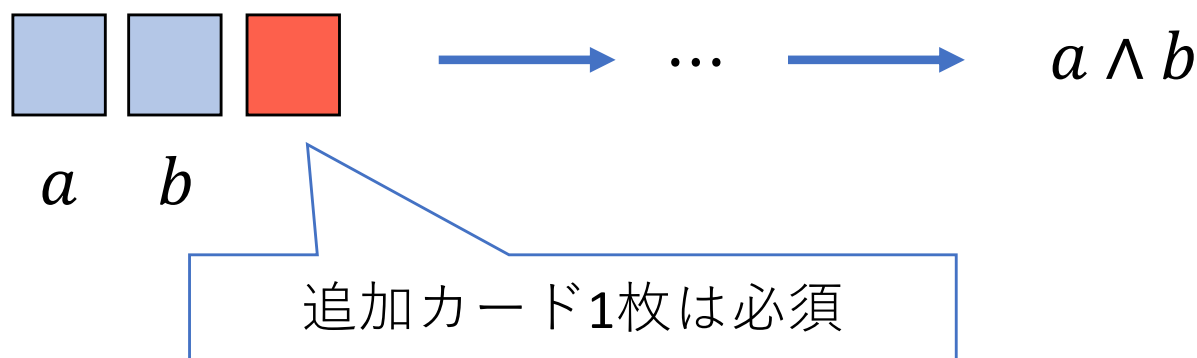
step d : (perm, $((\neg, \neg, \neg, \neg, \neg), (1\ 4)(2\ 3))$) if (turn, {5}) = $\{\uparrow\}$



1. 背景
 - i. カードベース暗号
 - ii. 上下カード版Five-Card Trick
 - iii. 貢献
2. 準備
 - i. しきい値関数
 - ii. 定式化モデル
3. 不可能性証明
4. $n + 1$ 枚 n 入力コミット型閾値プロトコル
5. 結論

不可能性証明

- 2枚2入力ANDプロトコルは存在しない



- この結果を3入力以上の場合にも適用し、一般的な n 枚 n 入力しきい値プロトコルの不可能性に拡大

構築

- $n + 1$ 枚 n 入力しきい値プロトコル

今後

- 同様の議論を対称関数への拡張