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Ⅱ

ランダムカットのみの４枚XORプロトコルに対する形式検証と不可能性証明

発表:池田 昇太†

共同研究者:藤田 和弘† 品川 和雅§‡ 米山 一樹†

†茨城大学 §筑波大学 ‡産業技術総合研究所

本講演の概要

- 形式検証を用いたカードベース暗号の不可能性証明について
- 4枚コミット型XORプロトコルの不可能性について

論文情報

ランダムカットのみの4枚XORプロトコルに対する形式検証と不可能性証明
*藤田和弘, 池田昇太, 品川和雅, 米山一樹
2025年暗号と情報セキュリティシンポジウム（SCIS2025), 福岡県小倉市, 2025年1月30日

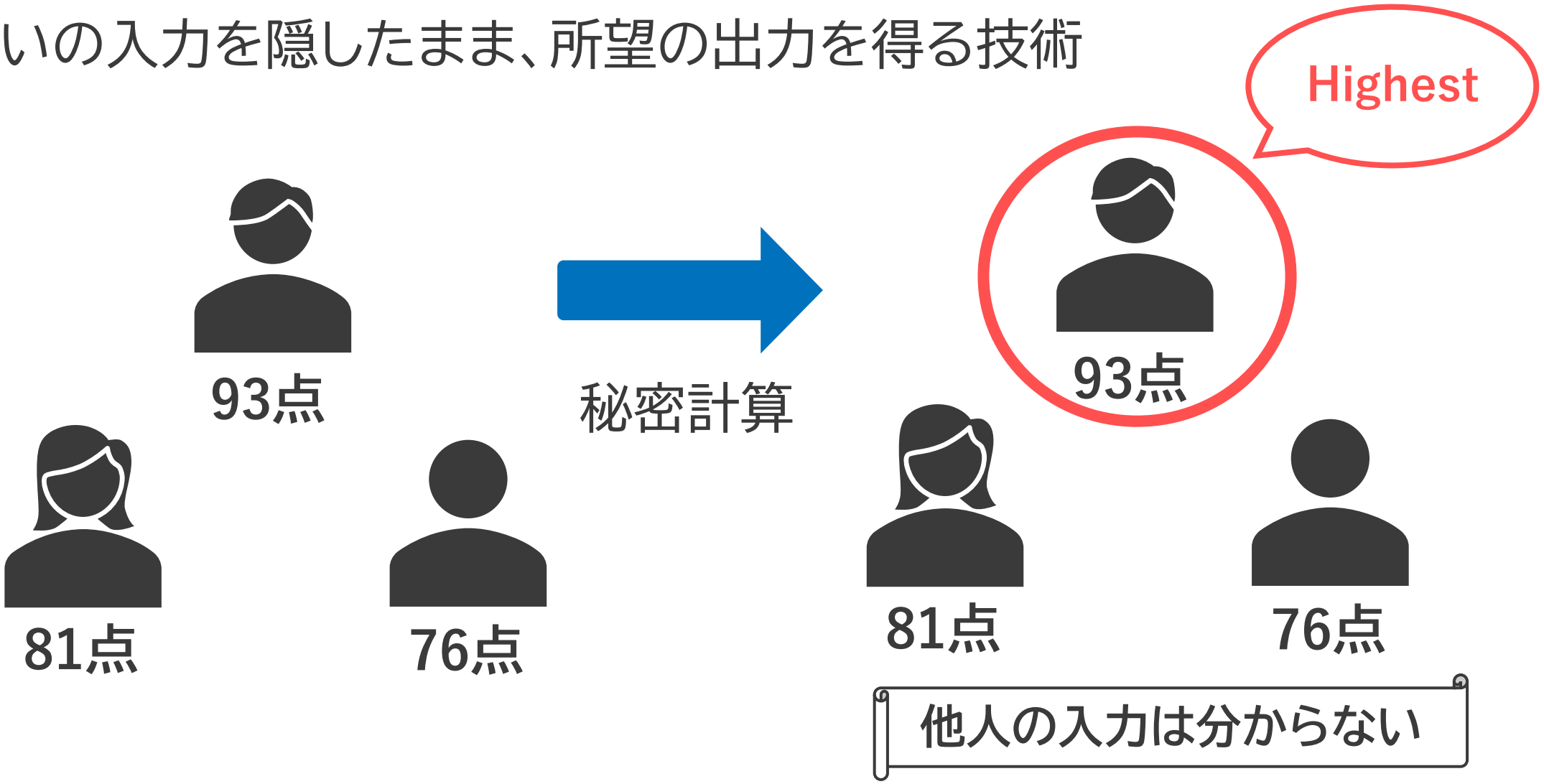
Formal Verification and Proof of Impossibility for Four-Card XOR Protocols Using Only Random Cuts.
Kazuhiro Fujita, *Shota Ikeda, Kazumasa Shinagawa, Kazuki Yoneyama
APKC 2025 (to appear)

目次

目次

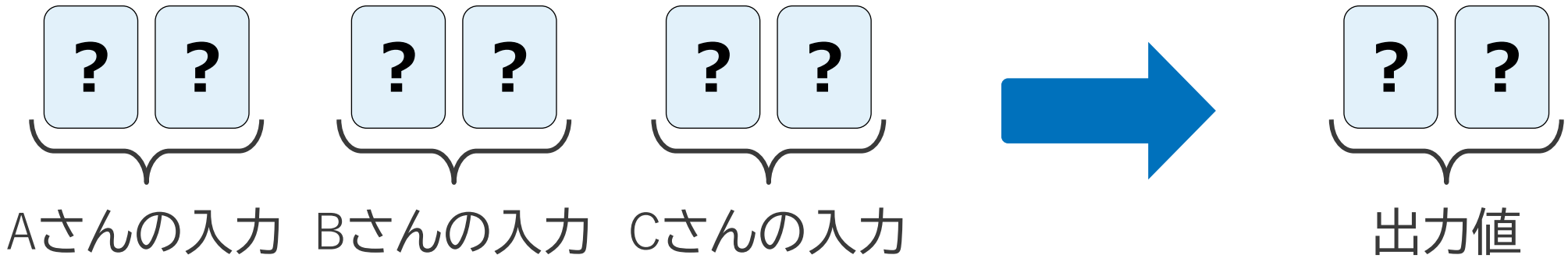
秘密計算

- 互いの入力を隠したまま、所望の出力を得る技術



カードベース暗号

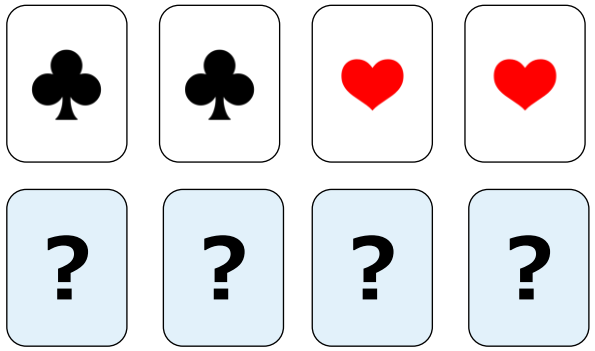
- 物理的なカードを用いて秘密計算を実現する技術



カードベース暗号の特徴

- 手軽に実演でき、仕組みを理解しやすい

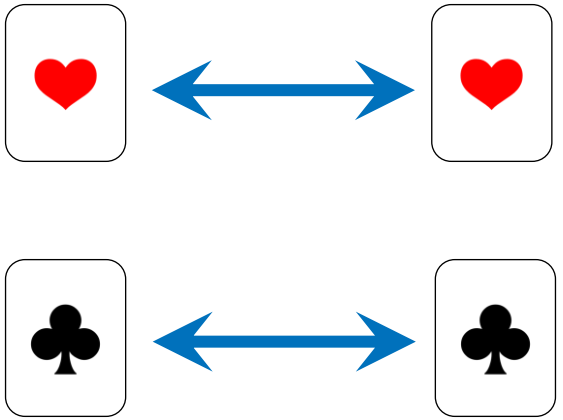
使用するカード



オモテ面

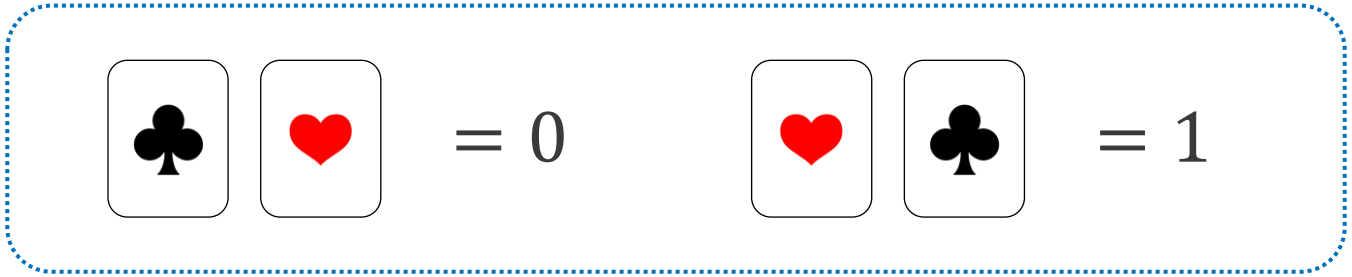
ウラ面

2色カード

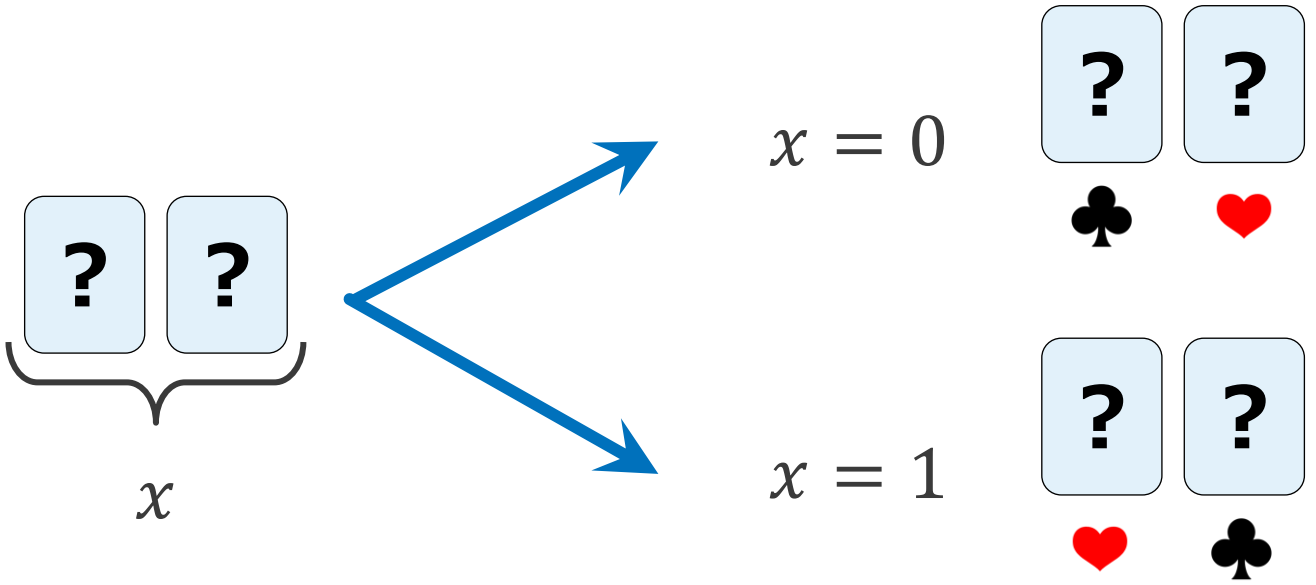


同じ色のカード同士は
全く区別がつかない

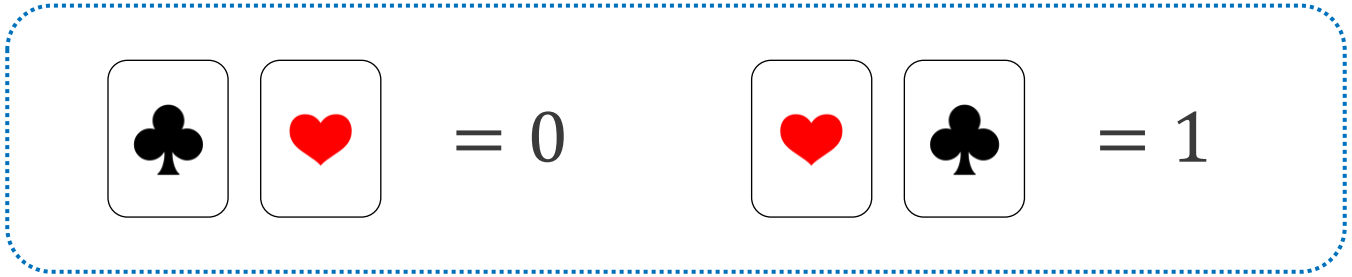
カードの符号化



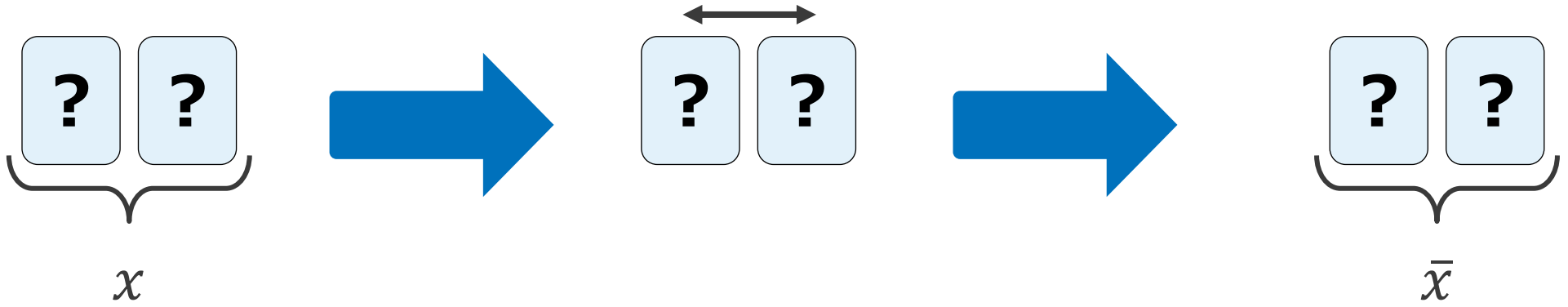
符号化に従う裏向きのカードをコミットメントと呼ぶ



コミットメントと否定

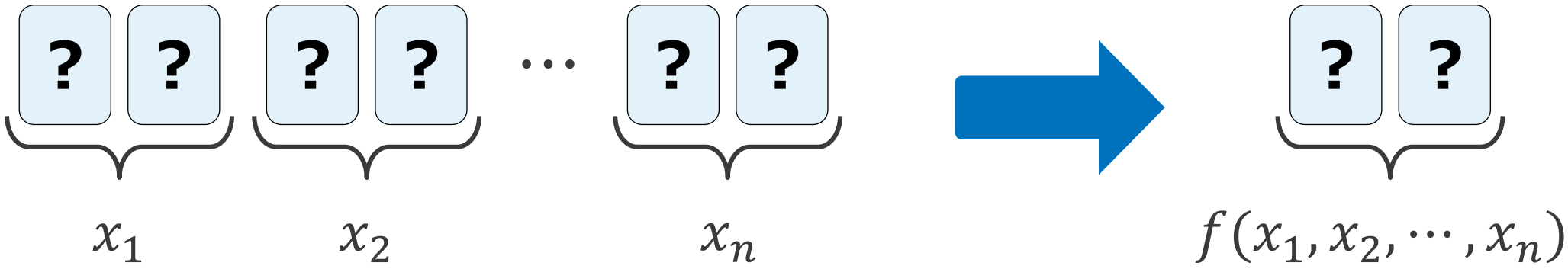


コミットメントの左右のカードを入れ替えることで否定(NOT)が容易に可能

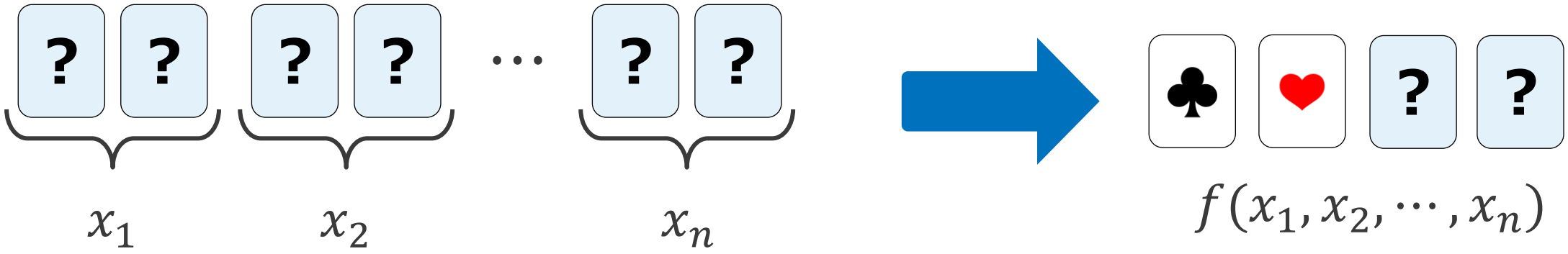


プロトコルの種類

- **コミット型プロトコル**:コミットメントを出力するプロトコル



- **非コミット型プロトコル**:出力値そのものを公開するプロトコル



プロトコルの種類

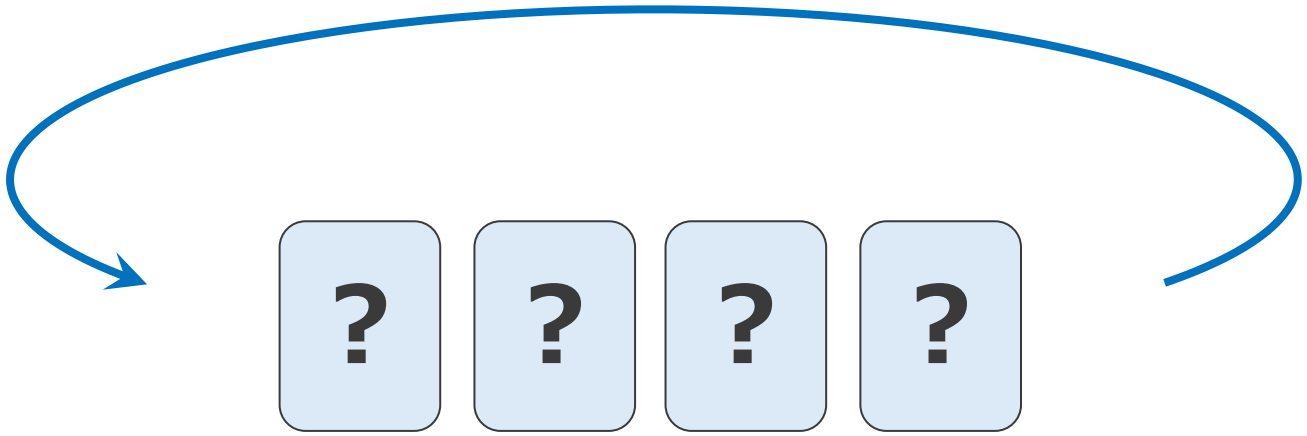
- **有限時間プロトコル**　：操作回数が有限回のプロトコル
- **Las Vegasプロトコル**：操作回数の期待値が有限回のプロトコル

→最悪の場合、プロトコルが終了しない

目次

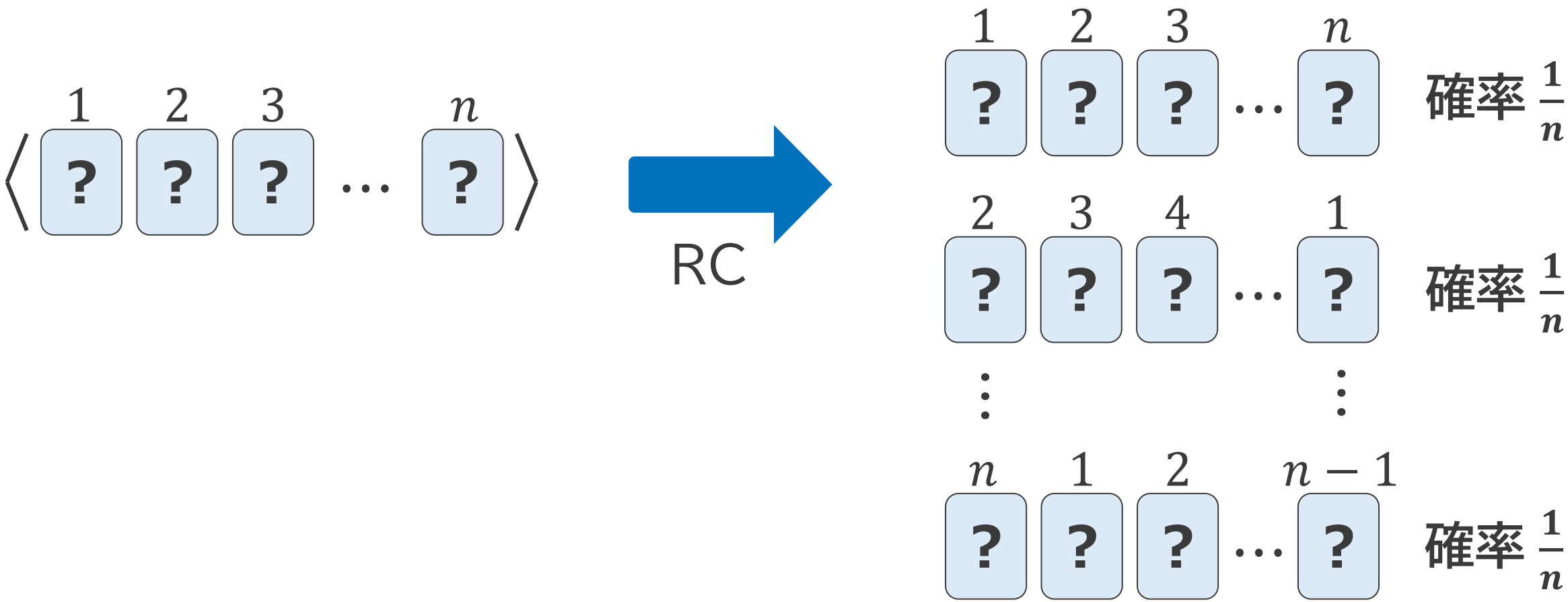
目次

ランダムカット(RC)



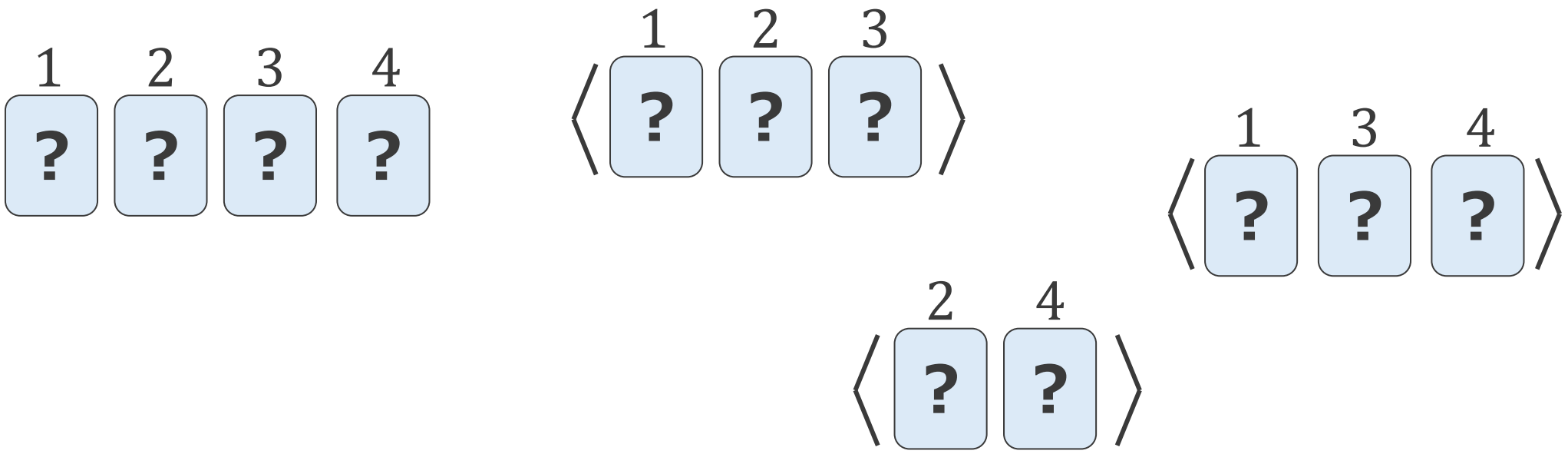
- カードを巡回的にランダムな回数並び替えるシャッフル
- カードベース暗号で最も基本的なシャッフル

ランダムカット(RC)



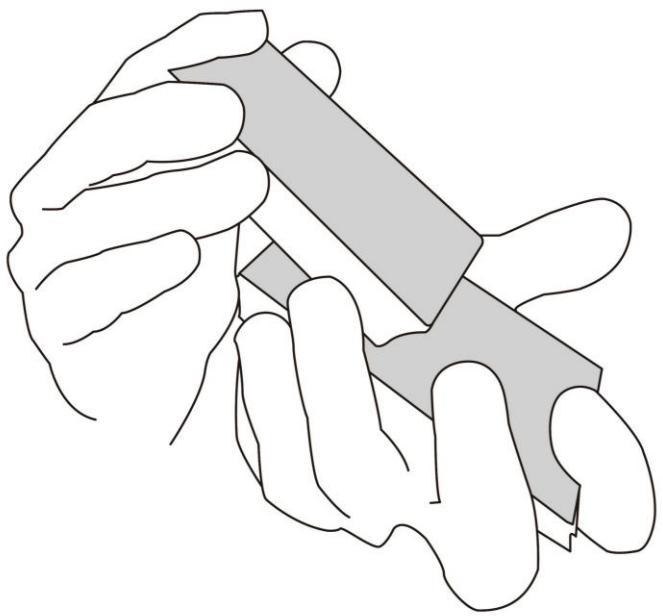
- n通りの巡回シフトが一様な確率で生じる
- どの並びになったのかは**誰にも**推測できない

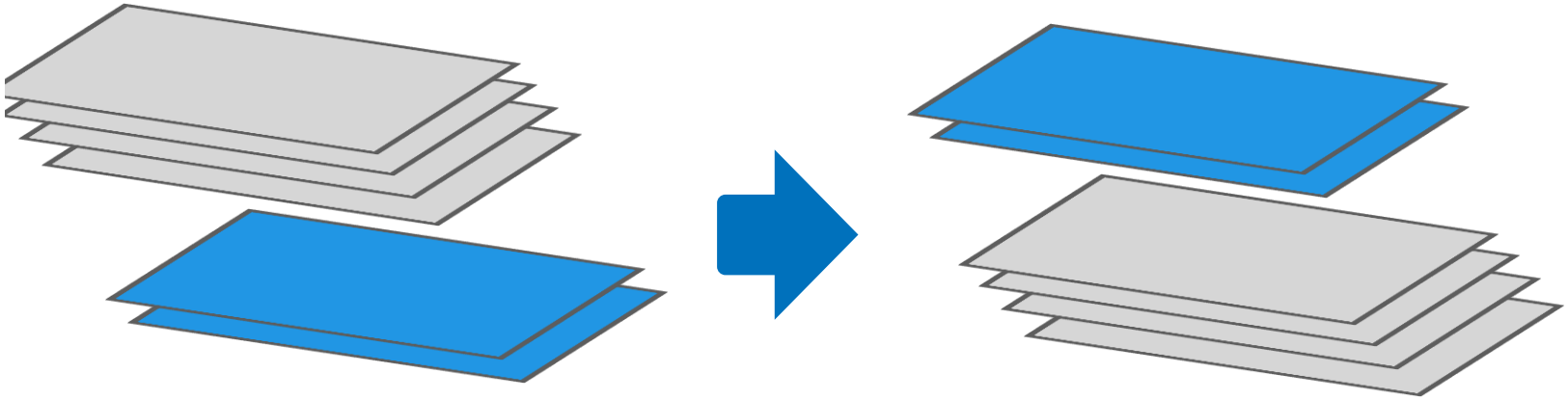
ランダムカット(RC)



- 4枚のカードすべてを巡回的にランダマイズするだけではない
- 部分的なカード列に対してもシャッフルとして適用できる

ランダムカットの実装





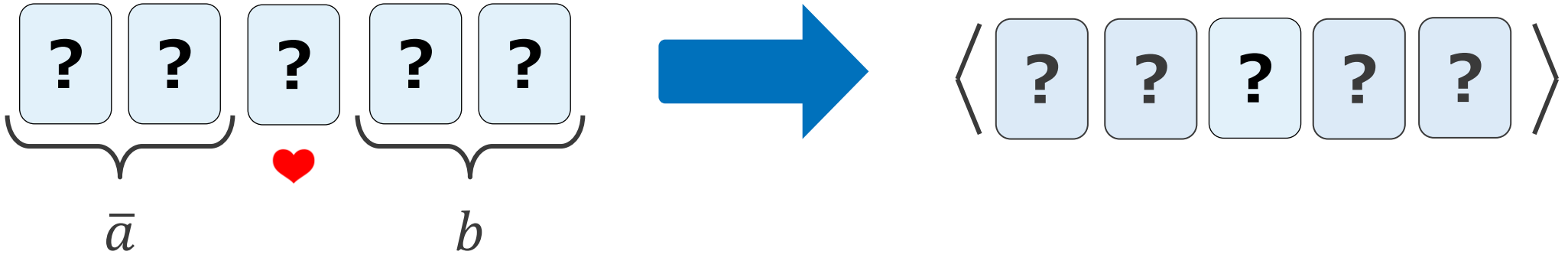
山札の下を取り出し　　山札の上に

- ヒンズーシャッフルで簡単に実装可能[UNHM+16]
- ランダムカットのみのプロトコルが好ましい

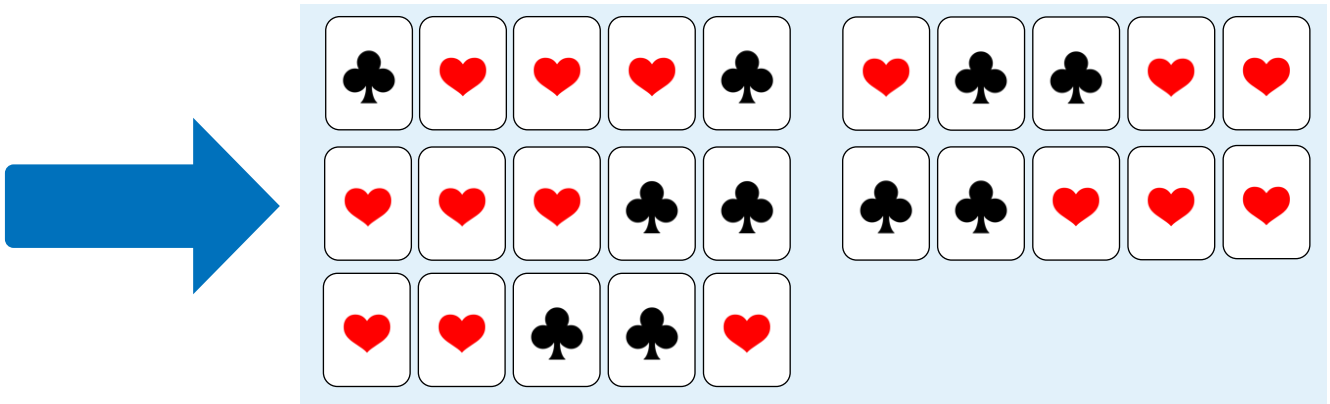
[UNHM+16] Ueda, I., Nishimura, A., Hayashi, Y. I., Mizuki, T., & Sone, H. How to implement a random bisection cut. TPNC 2016

RCのみの非コミット型ANDプロトコル

- 非コミット型ではRC1回でANDプロトコルが構成できる[DB90]

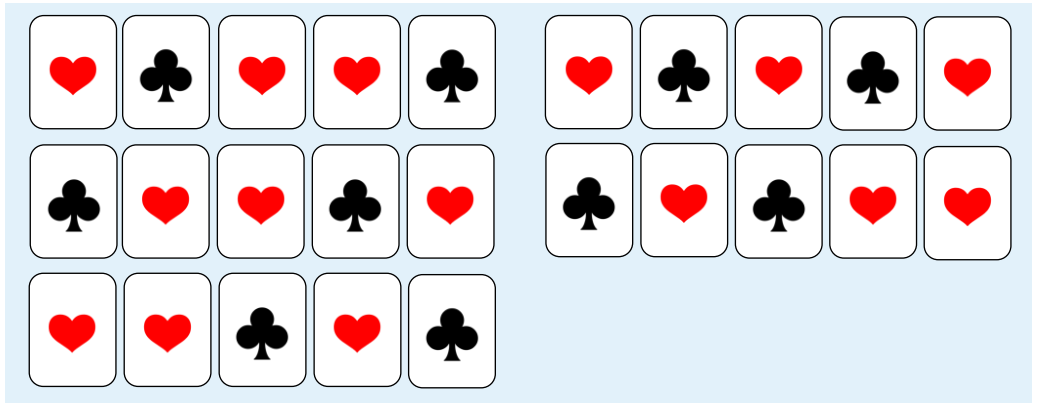


$a \wedge b = 1$



or

$a \wedge b = 0$



[DB90] Bert Den Boer. More efficient match-making and satisfiability the five card trick. Advances in Cryptology—EUROCRYPT ’89, 1990.

RCのみコミット型ANDプロトコル

	色数	枚数	有限時間	回数
Crépeau-Kilian[CK94]	4	10		8
Niemi-Renvall[NR98]	2	12		7.5
Stiglic[Sti01]	2	8		2
Abe et al.[AMS21]	2	6	✓	2

[CK94] C. Crépeau and J. Kilian, “Discreet solitary games,” CRYPTO ’93 1994.

[NR98] Valtteri Niemi and Ari Renvall. Secure multiparty computations without computers. TCS, 1998.

[Sti01] Anton Stiglic. Computations with a deck of cards. TCS, 2001.

[AMS21] Y. Abe, T. Mizuki, H. Sone, Committed-format AND protocol using only random cuts, Natural Computing 2021.

RCのみのコミット型AND未解決問題

未解決問題[水木23]

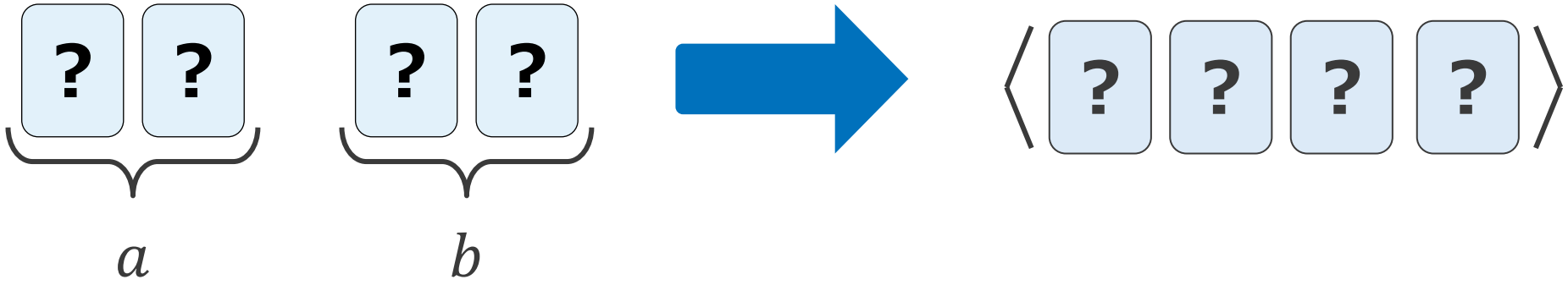
ランダムカットのみで5枚コミット型
ANDプロトコルが構成できるか？(Las Vegas)

- ランダムカットオンリーなため、考慮するべきシャッフルが少ない
- 存在するとしてもLas Vegasなので複雑なプロトコルになりそう？

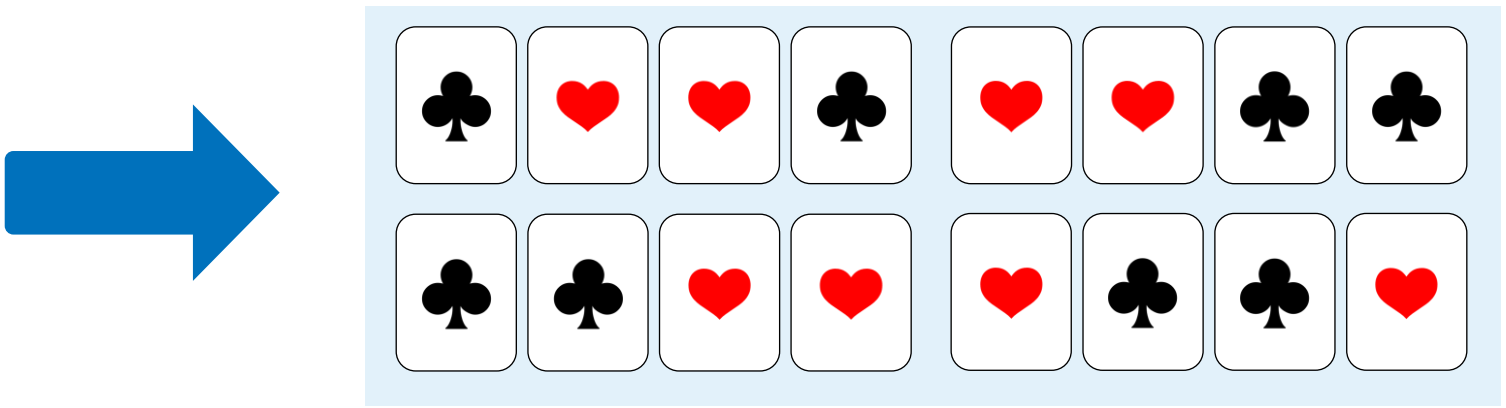
[水木23] 水木敬明. AND プロトコルにまつわる未解決問題. 産学連携によるカードベース暗号の数理的未解決問題と新課題の整理, 2023.

RCのみの非コミット型XORプロトコル

- 非コミット型ではRC1回でXORプロトコルが構成できる[SM18]

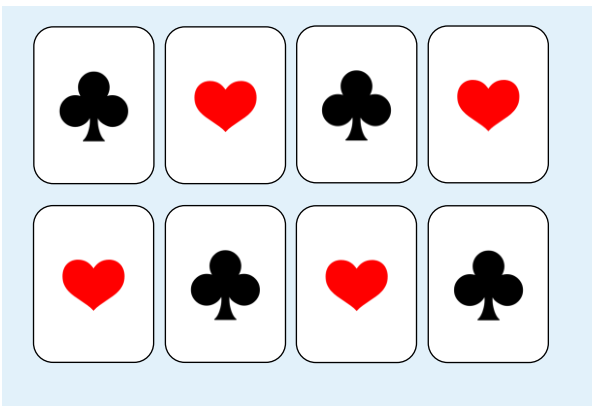


$a \oplus b = 1$



or

$a \oplus b = 0$



[SM18] Kazumasa Shinagawa, and Takaaki Mizuki. ”The six-card trick: secure computation of three-input equality.” ICISC, 2018.

RCのみコミット型XORプロトコル(1/2)

	色数	枚数	有限時間	回数
Crépeau-Kilian[CK94]	4	14		10
Mizuki et al.[MUS06]	2	10		7
Toyoda et al.[TMMS20]	2	6	✓	2

[CK94] C. Crépeau and J. Kilian, “Discreet solitary games,” CRYPTO ’93 1994.

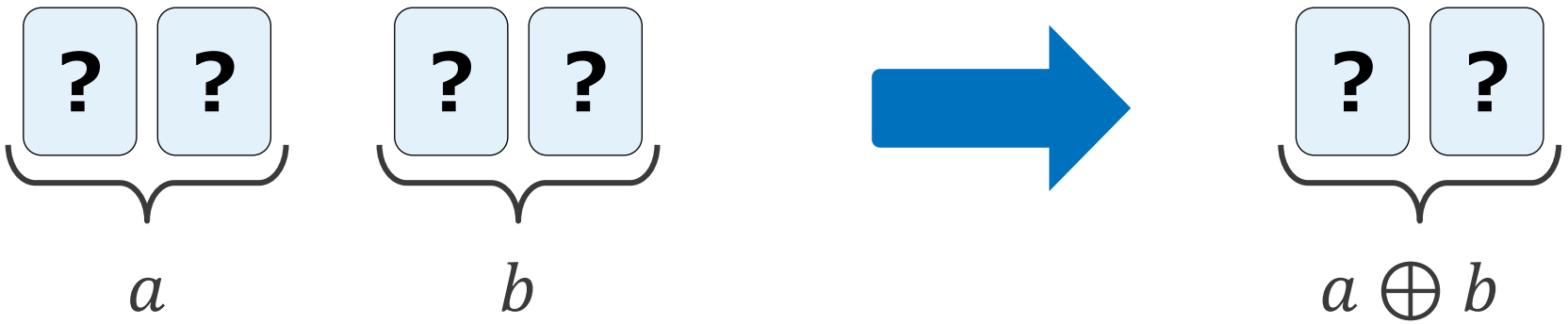
[MUS06] T. Mizuki, F. Uchiike, and H. Sone, “Securely computing XOR with 10 cards,” The Australasian Journal of Combinatorics, 2006.

[TMMS20] Kodai Toyoda, Daiki Miyahara, Takaaki Mizuki, Hideaki Sone.Six-card finite-runtime XOR protocol with only random cut, APKC.2020.

RCのみコミット型XORプロトコル(2/2)

枚数	プロトコル
◦ Secure XOR in a committed format using only Random Cuts	
6	Toyoda et al.[TMMS20]
5	？
4	？

- より少ない枚数でプロトコルが構成できないか？



RCのみのコミット型XOR未解決問題

未解決問題[TMMS20][水木23]

ランダムカットのみで5枚または4枚の
コミット型XORプロトコルが構成できるか？

- RCのみのため、考慮するべきシャッフルが少ない
- 4枚の場合は比較的簡単に示せそう？

[TMMS20] Kodai Toyoda, Daiki Miyahara, Takaaki Mizuki, Hideaki Sone.Six-card finite-runtime XOR protocol with only random cut, APKC2020.

[水木23] 水木敬明. AND プロトコルにまつわる未解決問題. 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, 2023.

本研究の貢献

- RCのみの4枚コミット型XORの不可能性を示した
  - 形式検証：操作4回以下の不可能性
  - 手証明　：一般の操作回数における不可能性
- 形式手法と手証明で互いの弱みを補完

枚数	プロトコル
◦ Secure XOR in a committed format using only Random Cuts	
6	Toyoda et al.[TMMS20]
5	？
4	×[Ours]

目次

目次

カードベース暗号と形式検証(1/3)

- Koch-Schrempp-Kirstenが初めてカードベース暗号に形式検証を用いる[KSK19]
  - ANDプロトコルの不可能性検証に形式手法を使用
  - C言語のプログラムに落とし込み検証

[KSK19] Koch, A., Schrempp, M. and Kirsten, M, Card-based cryptography meets formal verification, ASIACRYPT 2019

カードベース暗号と形式検証(2/3)

- Hoffが[KSK19]をOR、XOR、COPYに拡張[Hoff23]
  - 4枚コミット型XORプロトコルとしてMizuki-SoneのXORプロトコル※が発見できることを示した
  - 検証の高速化

※ランダム二等分割カット(RBC)1回のプロトコル

[Hoff23] Anne Elisabeth Hoff. An Automated Approach to Generating Card-BasedCryptographic Protocols. Bachelor's thesis.2023. https://github.com/a-nne-h/automatedApproachToGeneratingCardProtocols/tree/main

カードベース暗号と形式検証(3/3)

- 藤田らが[KSK19]のコードを修正[FYS24]
  - C言語のプログラムにミスがあった
  - 修正方法を提案

[FYS24] 藤田和弘, 米山一樹, 品川和雅. カードベース暗号の形式検証の再考. JSIAM2024

プロトコルで許される操作

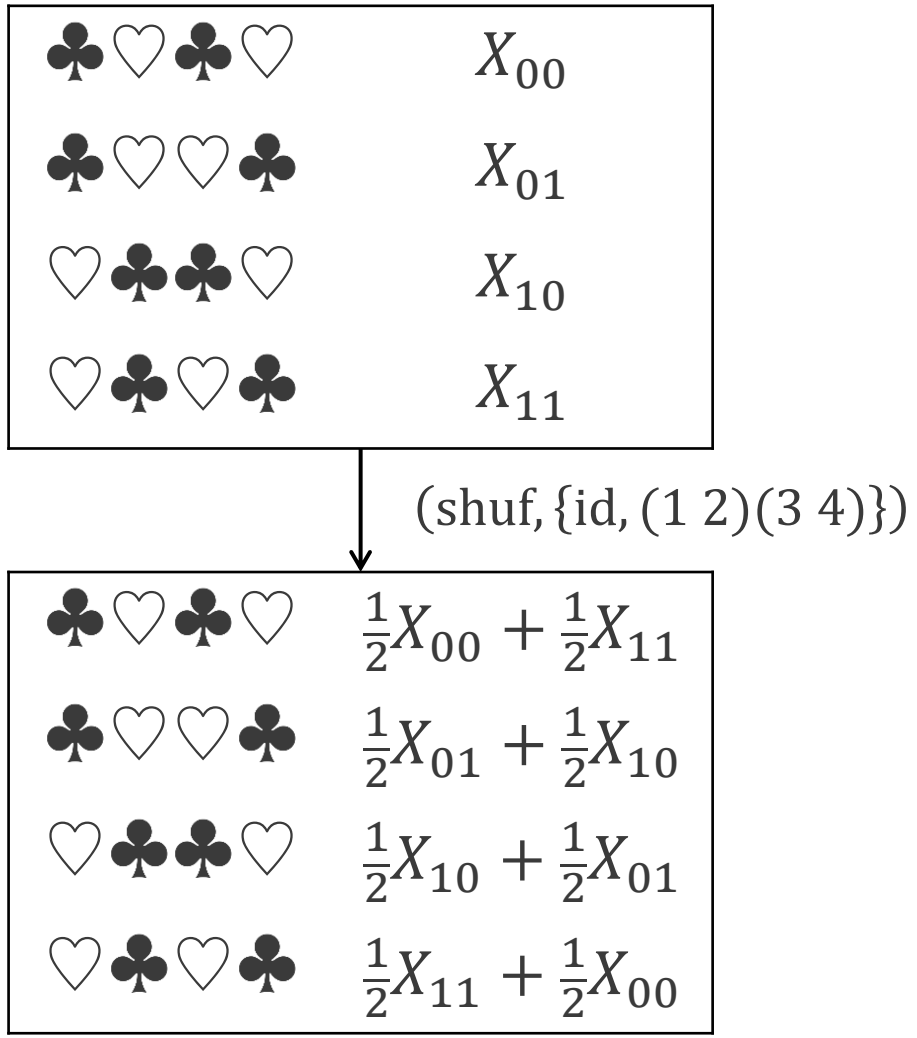
- プロトコルは抽象機械によって定式化されている[MS14]

- プロトコルで許されている操作は主に3つ
  - $(\mathrm{turn}, \mathrm{T})$ ：Tに含まれる位置番号のカードをめくる
  - (perm, π) ：置換πをカード列に適用
  - $(\mathrm{shuffle}, \Pi, \mathcal{F})$ ：分布$\mathcal{F}$に従い置換$\pi \in \Pi$を適用

[MS14] T. Mizuki and H. Shizuya, A formalization of card-based cryptographic protocols via abstract machine, Int. J. Inf. Sec. 2014

KWH-tree[KWH15]

- プロトコルをグラフとして図式化
- 図のボックスは**ステート**
- ステートは現在のカード列(の確率分布)
- $X_{a,b}$は各入力の確率変数

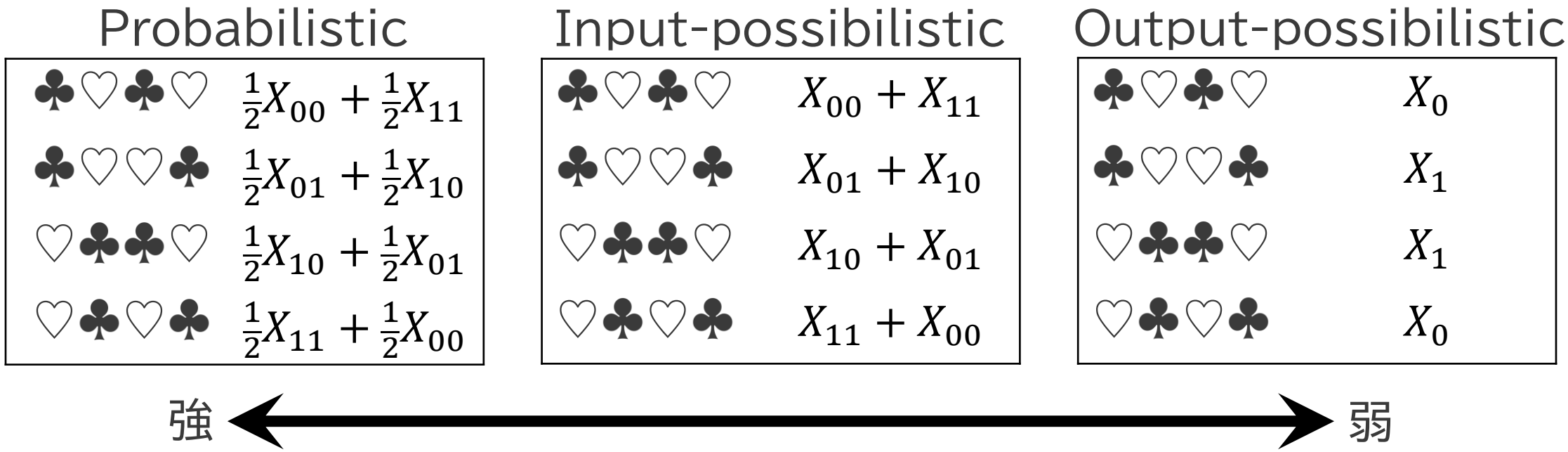


[KWH15] A. Koch, S. Walzer, and K. Härtel, Card-Based Cryptographic Protocols Using a Minimal Number of Cards, Asiacrypt 2015.

安全性レベル

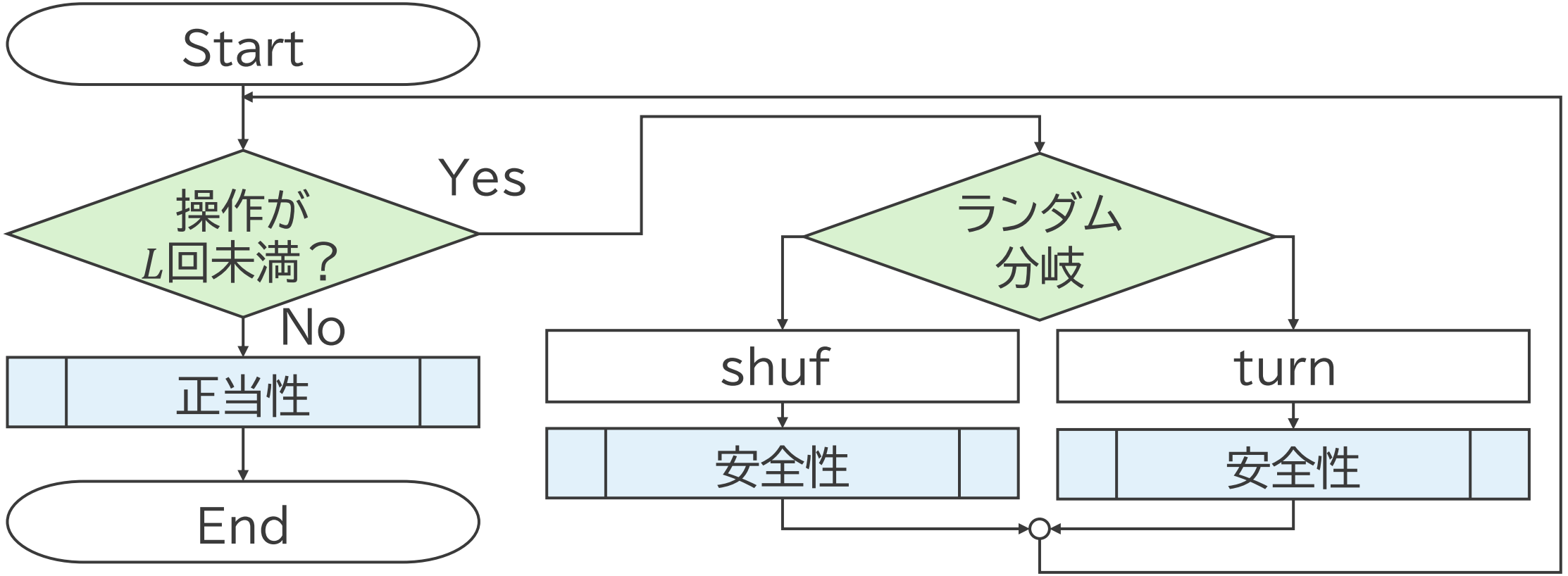
- Probabilistic security
  どの入力値も等確率で生じることを保証(通常の安全性レベル)
- Input-possibilistic security
  どの入力値も正の確率で生じることを保証
- Output-possibilistic security
  どの出力値も正の確率で生じることを保証

安全性レベル


Probabilistic
♣♡♣♡ $\frac{1}{2}X_{00} + \frac{1}{2}X_{11}$
♣♡♡♣ $\frac{1}{2}X_{01} + \frac{1}{2}X_{10}$
♡♣♣♡ $\frac{1}{2}X_{10} + \frac{1}{2}X_{01}$
♡♣♡♣ $\frac{1}{2}X_{11} + \frac{1}{2}X_{00}$
Input-possibilistic
♣♡♣♡ $X_{00} + X_{11}$
♣♡♡♣ $X_{01} + X_{10}$
♡♣♣♡ $X_{10} + X_{01}$
♡♣♡♣ $X_{11} + X_{00}$
Output-possibilistic
♣♡♣♡ X_0
♣♡♡♣ X_1
♡♣♣♡ X_1
♡♣♡♣ X_0
強
弱

不可能性検証の流れ(1/2)

- プロトコル・安全性・正当性をC言語で記述



不可能性検証の流れ(2/2)

- CBMC[CKL04]に入力し検証する

[CKL04] E. Clarke, D. Kroening, and F. Lerda. A Tool for Checking ANSI-C Programs. TACAS 2004.

検証を行う安全性レベル

- Output-possibilisticでは“turn”1回のみのXORが存在
- そのため検証はInput-possibilisticで行う

Probabilistic		Input-possibilistic		Output-possibilistic	
♣♡♣♡	$\frac{1}{2}X_{00}+\frac{1}{2}X_{11}$	♣♡♣♡	$X_{00}+X_{11}$	♣♡♣♡	X_0
♣♡♡♣	$\frac{1}{2}X_{01}+\frac{1}{2}X_{10}$	♣♡♡♣	$X_{01}+X_{10}$	♣♡♡♣	X_1
♡♣♣♡	$\frac{1}{2}X_{10}+\frac{1}{2}X_{01}$	♡♣♣♡	$X_{10}+X_{01}$	♡♣♣♡	X_1
♡♣♡♣	$\frac{1}{2}X_{11}+\frac{1}{2}X_{00}$	♡♣♡♣	$X_{11}+X_{00}$	♡♣♡♣	X_0

強 ⟷ 弱

“turn”1回のXORの直感

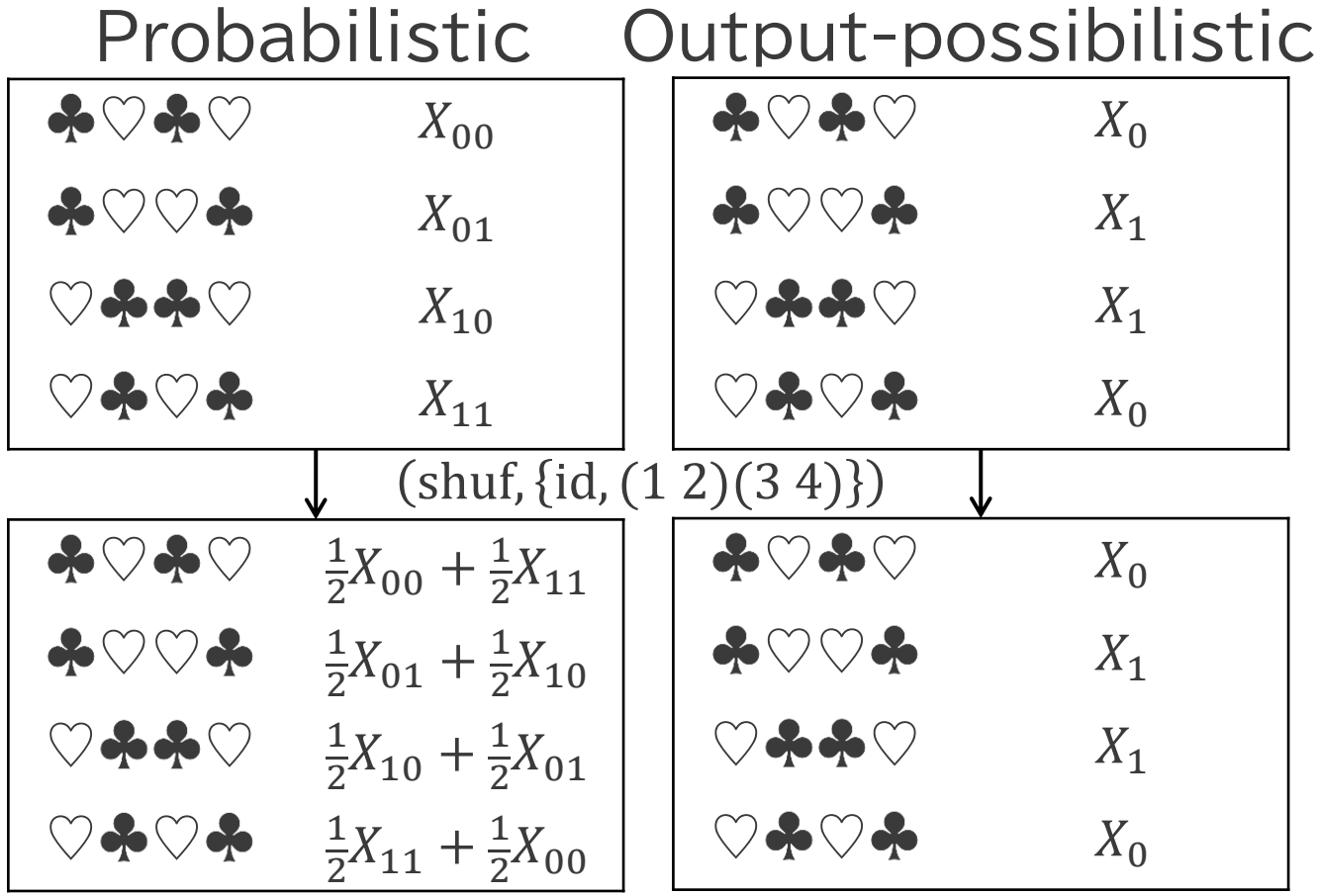
- Output-possibilisticでは、
  “turn”1回のみでXORプロトコルが構成可能

- 直感:　Mizuki-SoneのXORプロトコル[MS09]を
  Output-possibilisticにしたもの

[MS09] T. Mizuki and H. Sone. Six-card secure AND and four-card secure XOR. FAW 2009.

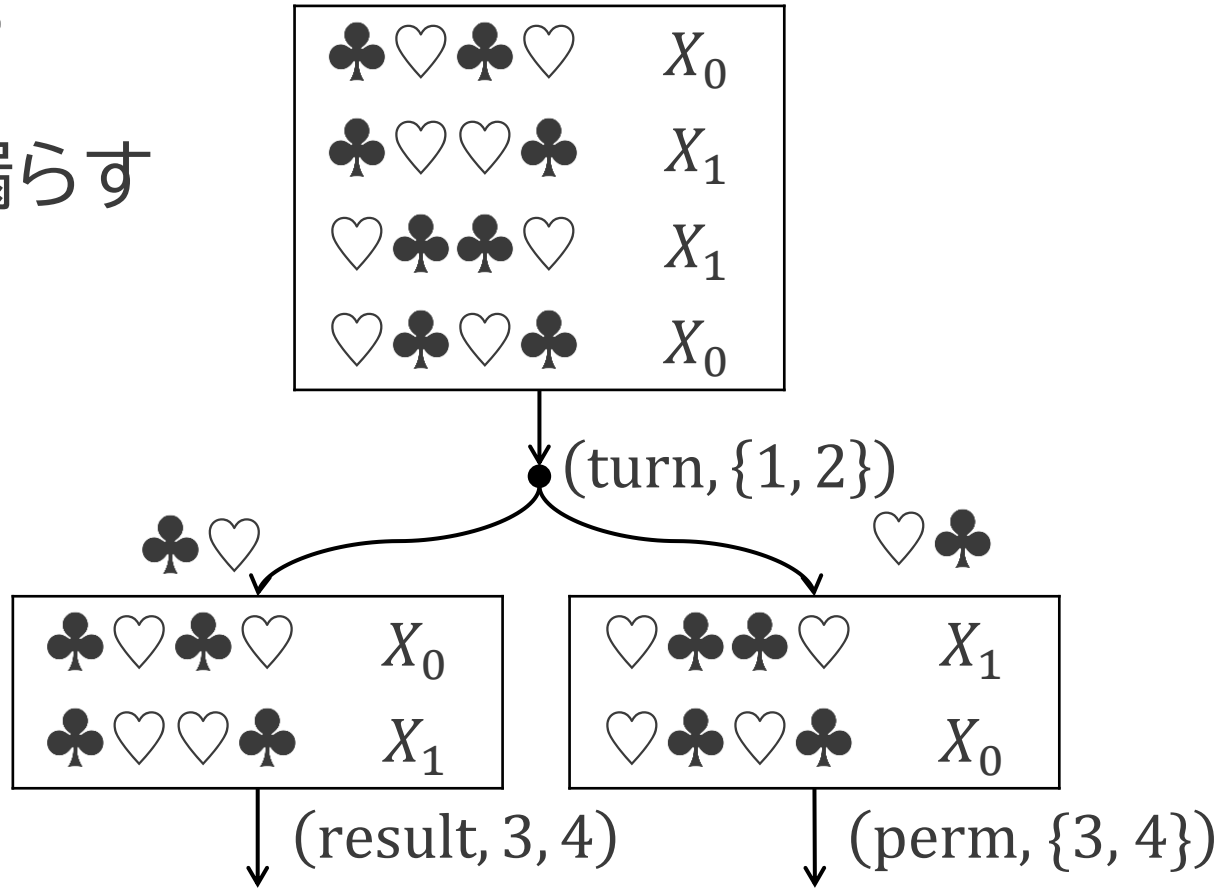
Mizuki-Soneのshufとstate

- Output-possibilisticでは最初のshufが冗長な操作になる



turn1回のXOR

- [MS09]はOutputでは初期状態から直接turnで構成可能
- turn1回のXORになっている
- ただし、これでは入力情報を漏らす



検証環境

- Intel Core i9-11900K
- 128GiB RAM搭載 Ubuntu 24.04 LTSマシン

形式検証の結果

- 操作が4回までの場合、構成不可能
  - RC2回以下の不可能性が言える
- 操作が5回の検証はメモリ不足により中断

操作回数[回]	不可能性	実行時間[s]
1	✓	157
2	✓	1092
3	✓	4091
4	✓	8205
5		11914

目次

目次

不可能性証明の概要

- 初期状態から適用可能な操作を適用しても、**最終状態に到達不可能**
- 適用可能な操作としては、**RCとturnのみ**
  - permを考えなくてよいことも示す

♣♡♣♡	X_{00}
♣♡♡♣	X_{01}
♡♣♣♡	X_{10}
♡♣♡♣	X_{11}

初期state（初期状態）

♣♡♣♡	$\frac{1}{2}X_{00} + \frac{1}{2}X_{11}$
♣♡♡♣	$\frac{1}{2}X_{01} + \frac{1}{2}X_{10}$
♡♣♣♡	$\frac{1}{2}X_{10} + \frac{1}{2}X_{01}$
♡♣♡♣	$\frac{1}{2}X_{11} + \frac{1}{2}X_{00}$

最終stateの例（最終状態）

再掲

プロトコルで許される操作

- プロトコルは抽象機械によって定式化されている[MS14]

- プロトコルで許されている操作は主に3つ
  - $(\mathrm{turn}, \mathrm{T})$ ：Tに含まれる位置番号のカードをめくる
  - (perm, π) ：置換πをカード列に適用
  - $(\mathrm{shuffle}, \Pi, \mathcal{F})$ ：分布$\mathcal{F}$に従い置換$\pi \in \Pi$を適用

[MS14] T. Mizuki and H. Shizuya, A formalization of card-based cryptographic protocols via abstract machine, IJIS. 2014

置換操作permの省略

置換操作permは操作として考慮しなくてもよい

入れ替え前	入れ替え後
$(\mathrm{perm}, \pi) \rightarrow (\mathrm{shuf}, \mathrm{RC}_{\sigma})$	$(\mathrm{shuf}, \mathrm{RC}_{\pi^{-1}\sigma\pi}) \rightarrow (\mathrm{perm}, \pi)$
$(\mathrm{perm}, \pi) \rightarrow (\mathrm{turn}, \{i\})$	$(\mathrm{turn}, \{\pi^{-1}(i)\}) \rightarrow (\mathrm{perm}, \pi)$

※RC_{σ}はσから生成されるRC

- 任意のRCと任意のターン位置を考慮するため、permは不要
- 任意のプロトコルはpermを用いないプロトコルに変換可能

ステートの定義

カード列の集合

$S := \{♡♣♡♣,\ ♡♣♣♡,\ ♣♡♡♣,\ ♣♡♣♡,\ ♡♡♣♣,\ ♣♣♡♡\}$

から非負係数の1次斉次多項式の集合

$$P := \{p_{11}X_{11}+p_{10}X_{10} + p_{01}X_{01} + p_{00}X_{00} \mid p_{ij} \geq 0\}$$

への写像$\mu : S \rightarrow P$として定義

♣♡♣♡	$\frac{1}{2}X_{00} + \frac{1}{2}X_{11}$
♣♡♡♣	$\frac{1}{2}X_{01} + \frac{1}{2}X_{10}$
♡♣♣♡	$\frac{1}{2}X_{10} + \frac{1}{2}X_{01}$
♡♣♡♣	$\frac{1}{2}X_{11} + \frac{1}{2}X_{00}$

ステートのturnability

ステートμが$i \in \{1,2,3,4\}$についてturnableであるとは

任意の$c \in \{\heartsuit, \clubsuit\}$ に対して以下が成り立つこと

$$\sum_{s \in S, s[i]=c} \mu(s) = p(X_{00} + X_{01} + X_{10} + X_{11})$$

i ：位置番号（カード列の左端のカードが1番目）

$s[i]$ ：カード列sのi番目のシンボル

⊥-sequence

コミット型XORプロトコルでは最終的に

X_{00}, X_{11}とX_{01}, X_{10}が分離されている必要がある

逆に

ステートμがあるカード列$s \in S$で以下が成り立つ場合
プロトコルが正常に終了しない(**プロトコル構成不可**)

$$(*)\ \mu(s) = \sum_{a,b} p_{ab} X_{ab}\ (p_{00} + p_{11} > 0, p_{01} + p_{10} > 0)$$

条件$(*)$を満たすカード列$s \in S$を**⊥-sequence**と呼ぶ

適用可能なRC

現在のstateμに対してRC_σがμに対して**適用可能**とは

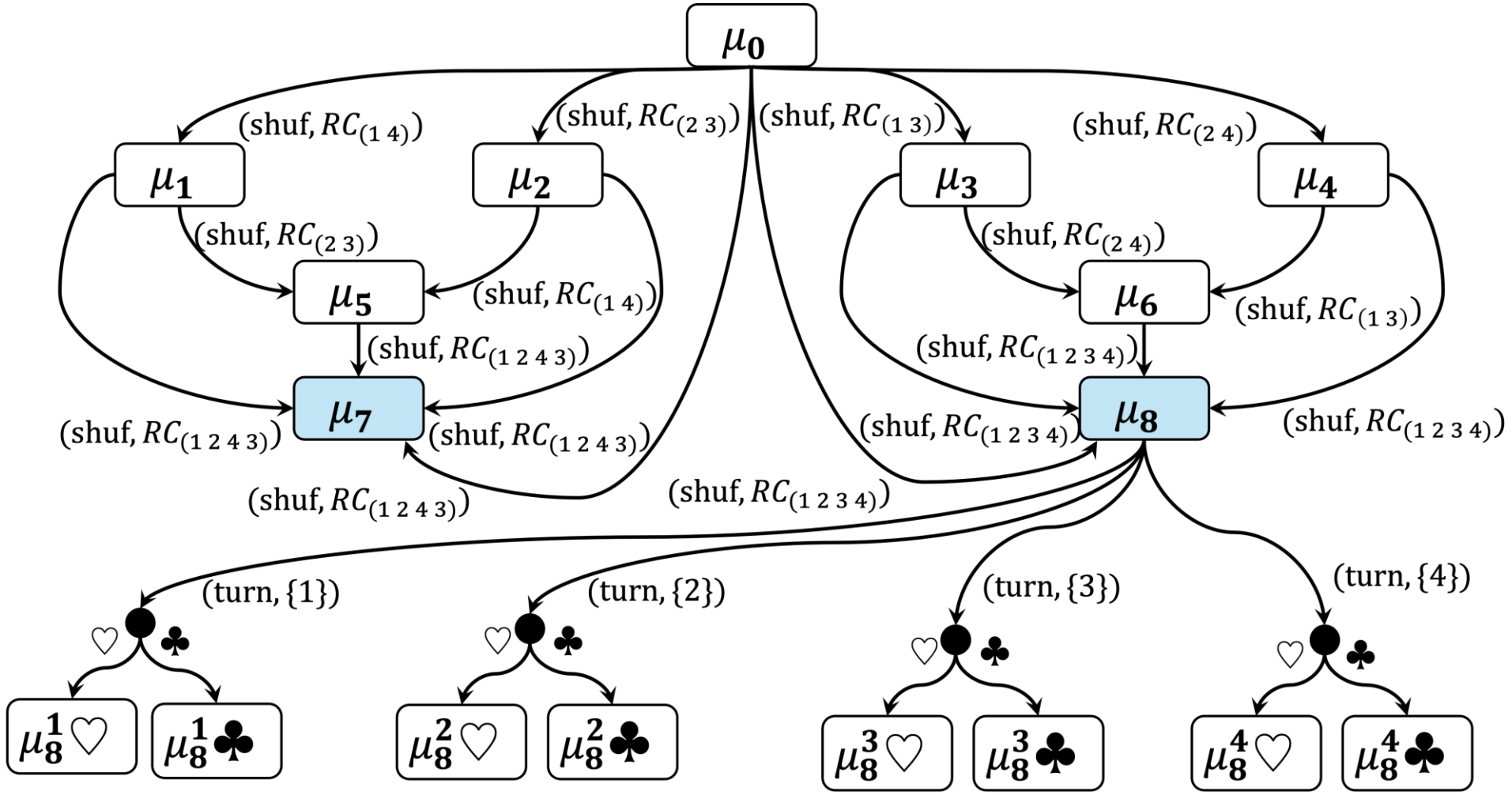
μにRC_σを適用して得られるstateμ'が以下の2条件を満たす

- stateμ'は⊥-sequenceを含まない
- stateμ'はstateμとは異なるstateである

不可能性証明の戦略

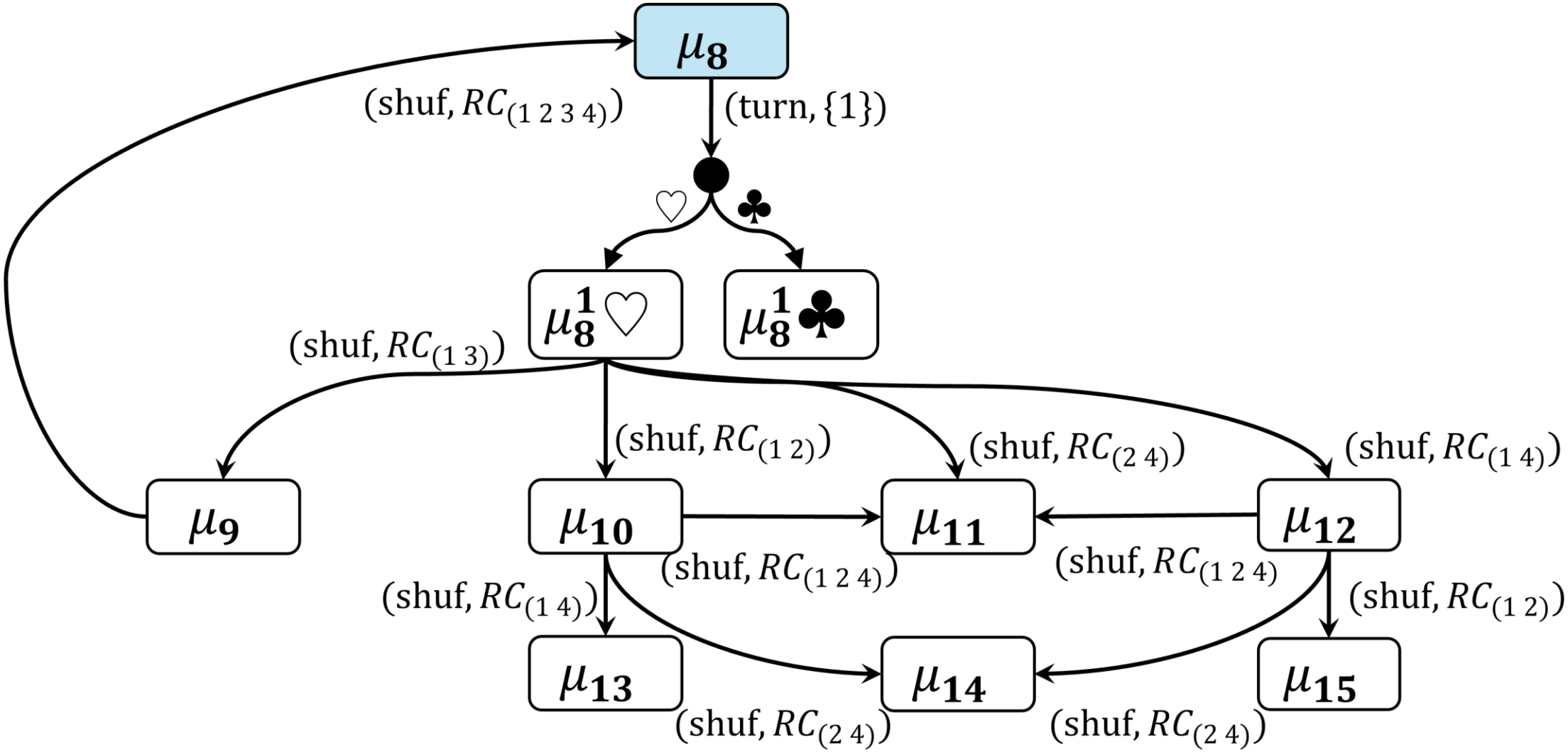
- stateμがカード位置iで**turnability**を満たすとき、
  stateμに対して$(\mathbf{turn}, \{i\})$を適用
- stateμにランダムカットRC_σが**適用可能**であるとき、
  stateμに対して$(\mathbf{shuf}, \mathbf{RC}_\sigma)$を適用

不可能性証明(1/2)



白色:turnableでないstate, 青色:turnableなstate

不可能性証明(2/2)


μ_8
(shuf, $RC_{(1\,2\,3\,4)}$)
(turn, {1})
♡
♣
μ_8^1♡
μ_8^1♣
(shuf, $RC_{(1\,3)}$)
(shuf, $RC_{(1\,2)}$)
(shuf, $RC_{(2\,4)}$)
(shuf, $RC_{(1\,4)}$)
μ_9
μ_{10}
μ_{11}
μ_{12}
(shuf, $RC_{(1\,2\,4)}$)
(shuf, $RC_{(1\,2\,4)}$)
(shuf, $RC_{(1\,4)}$)
(shuf, $RC_{(1\,2)}$)
μ_{13}
μ_{14}
μ_{15}
(shuf, $RC_{(2\,4)}$)
(shuf, $RC_{(2\,4)}$)

まとめ

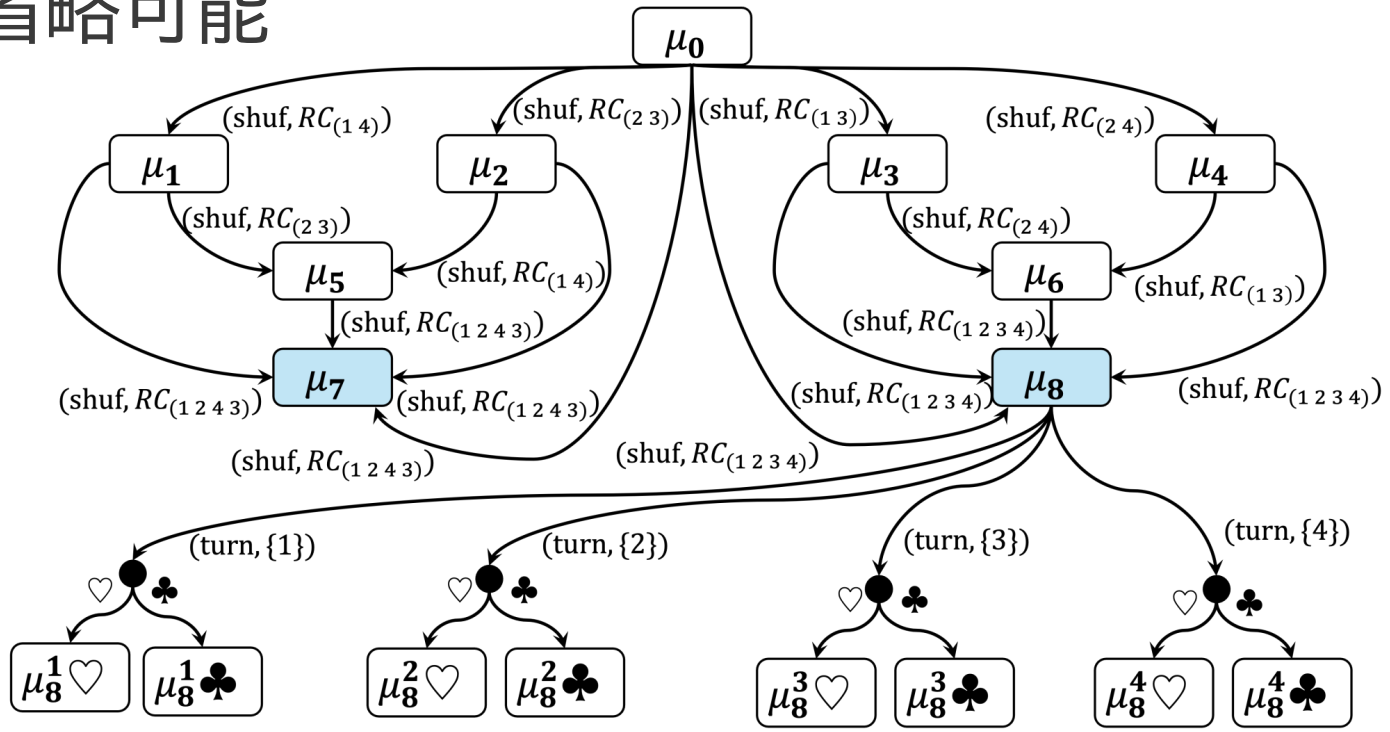
- RCのみの4枚コミット型XORプロトコルの不可能性を示した
  - 形式検証　：　操作4回以下（RC2回）の不可能性
  - 手証明　　：　一般の操作回数における不可能性

- 今後の課題
  - RCのみの5枚コミット型XORの不可能性証明

Appendix

不可能性証明

- μ_7はμ_8に変換可能なため、省略可能
- turn位置は$(\mathbf{turn}, \{\mathbf{1}\})$のみを考慮しても一般性を失わない
- ♣は♡に変換可能なため、省略可能
- そのため、μ_8を調べる



Appendix

不可能性証明

- 得られたどのstateも最終stateではない
- 最終的に得られたstateはどのturn位置でもturnabilityを満たさない
- 最終的に得られたstateには適用可能なRC_σはない