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

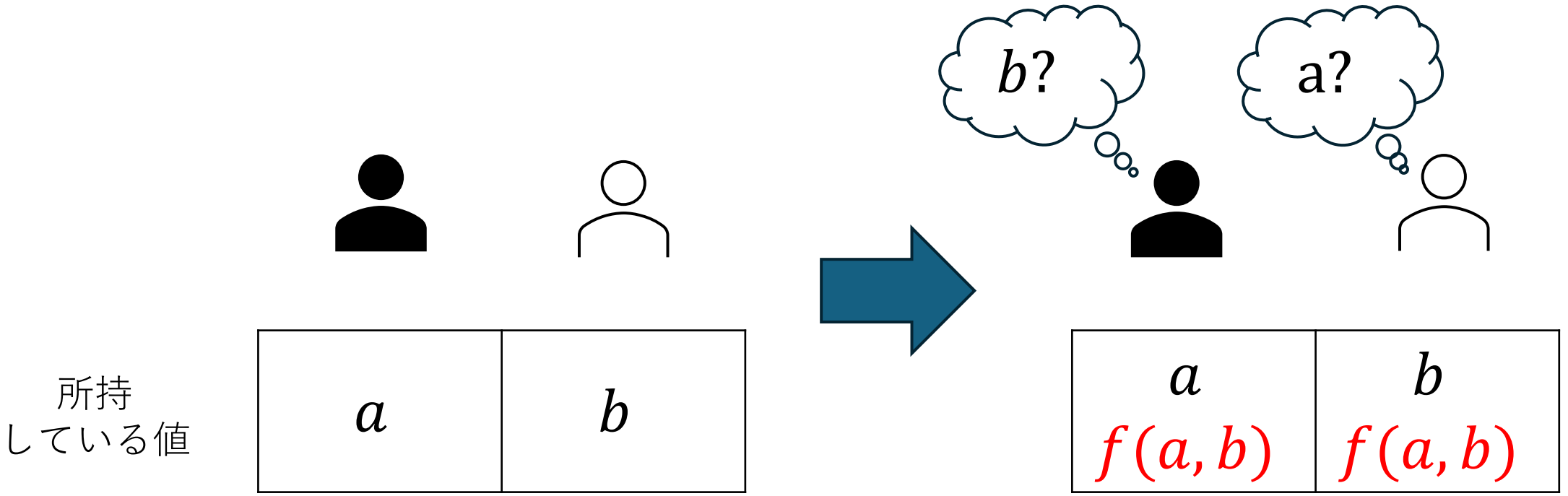
部分開示操作を用いた効率的なカードベースプロトコル

茨城大学　本多由昂

目録

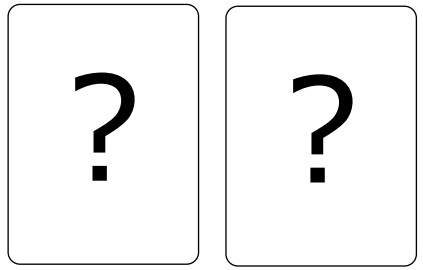
秘密計算

入力を秘匿したまま計算をする技術



カードベース暗号

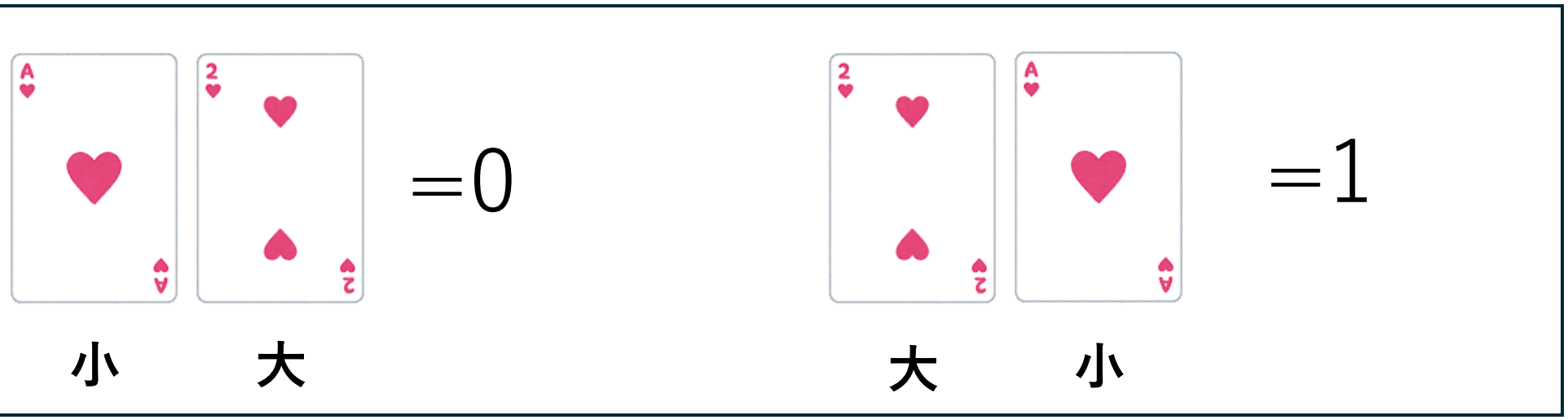
物理的なカード組を用いて秘密計算を行う技術



コミットメント

トランプカードを使用するプロトコル

Niemi-Renvall によって提案された[NR99]



・トランプカードは市販されているため手に入りやすい
・二色カードのプロトコルをそのまま実装することはできない

[NR99] Niemi and Renvall. Solitaire zero-knowledge. Fundam. Info. 1999.

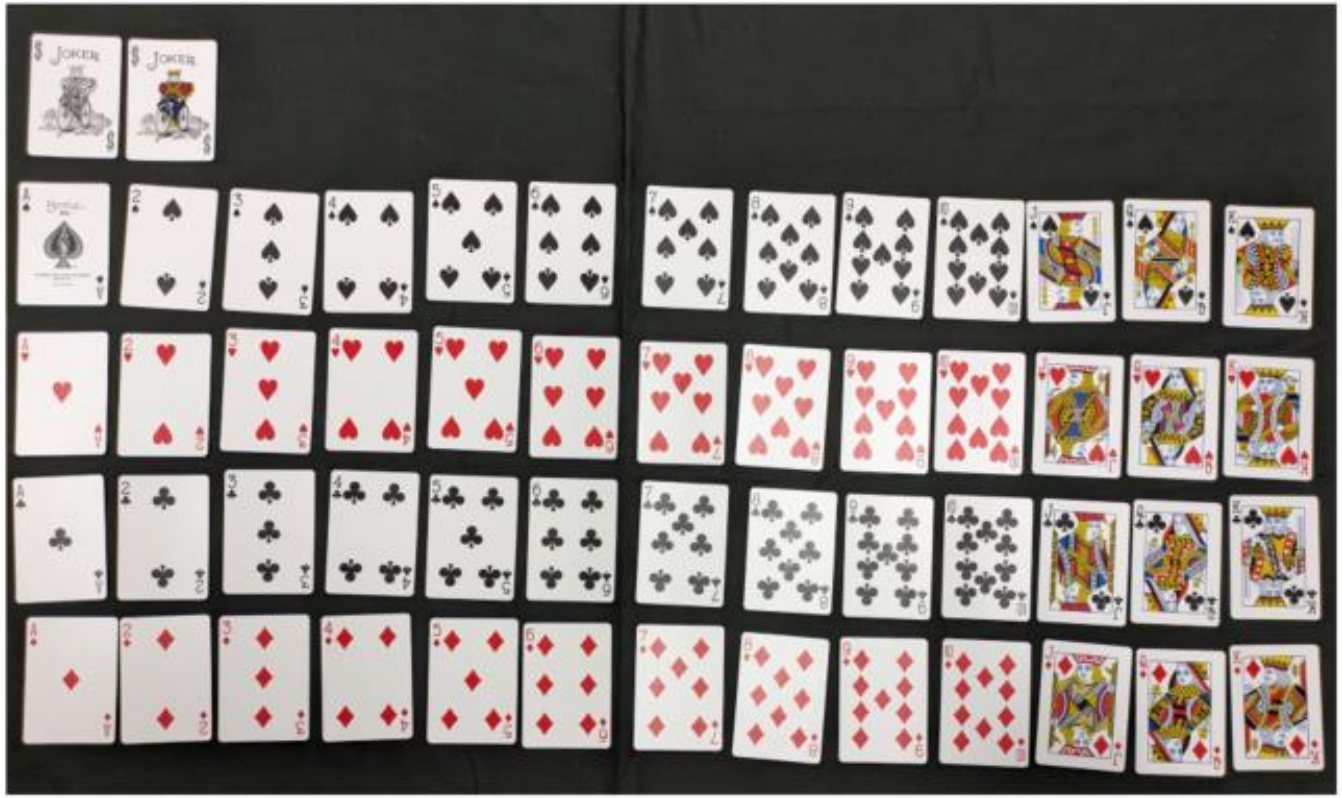
トランプカードの既存研究と本研究

	カード枚数	有限時間か？	シャッフル回数	シャッフルの種類	部分開示操作
コミット型ANDプロトコル					
[NR99]	5		9.5(exp.)	ランダムカット	不要
[Mizuki16]	8	✓	4	二等分割カット	不要
[KSK19]	4		6(exp.)	ランダムカット	不要
[HS24]	**4**	✓	**3**	**ランダムカット**	**必要**
COPYプロトコル					
[NR99]	6		5.5(exp.)	ランダムカット	不要
[Mizuki16]	6	✓	1	二等分割カット	不要
[HS24]	**4**	✓	3	**ランダムカット**	**必要**

[NR99] Niemi and Renvall. Solitaire zero-knowledge. Fundam. Info. 1999.
[KSK19] Koch, Schrempp, and Kirsten. Card-based cryptography meets formal verification. ASIACRYPT 2019.
[Mizuki16] Mizuki. Efficient and Secure Multiparty Computations Using a Standard Deck of Playing Cards. CANS2016.
[HS24] Honda,Shinagawa. Efficient Card-Based Protocols with a Standard Deck of Playing Cards Using Partial Opening. IWSEC 2024.

使用するカード

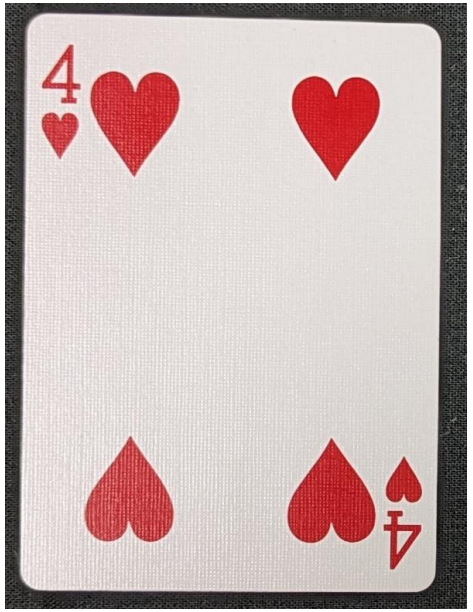
U.S.プレイングカード社 “バイスクル”



同社の“タリホー”や“ビー”でも本研究のプロトコルは実装可能

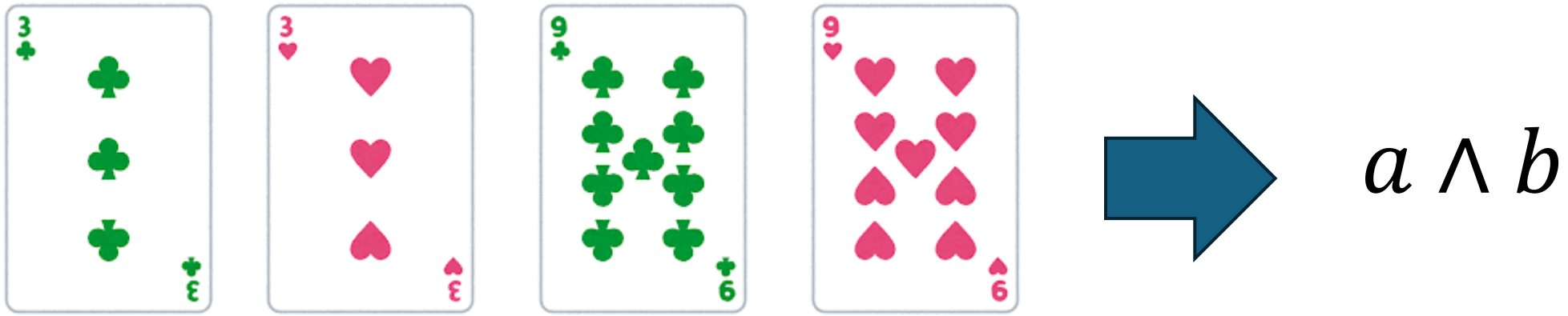
MiyaharaとMizukiの半開示操作

トランプカードの**スートのみ**を開示する操作
この操作による効率的な非コミット型ANDが提案された



[MM22] D. Miyahara and T. Mizuki. Secure computations through checking suits of playing cards. FAW 2022.

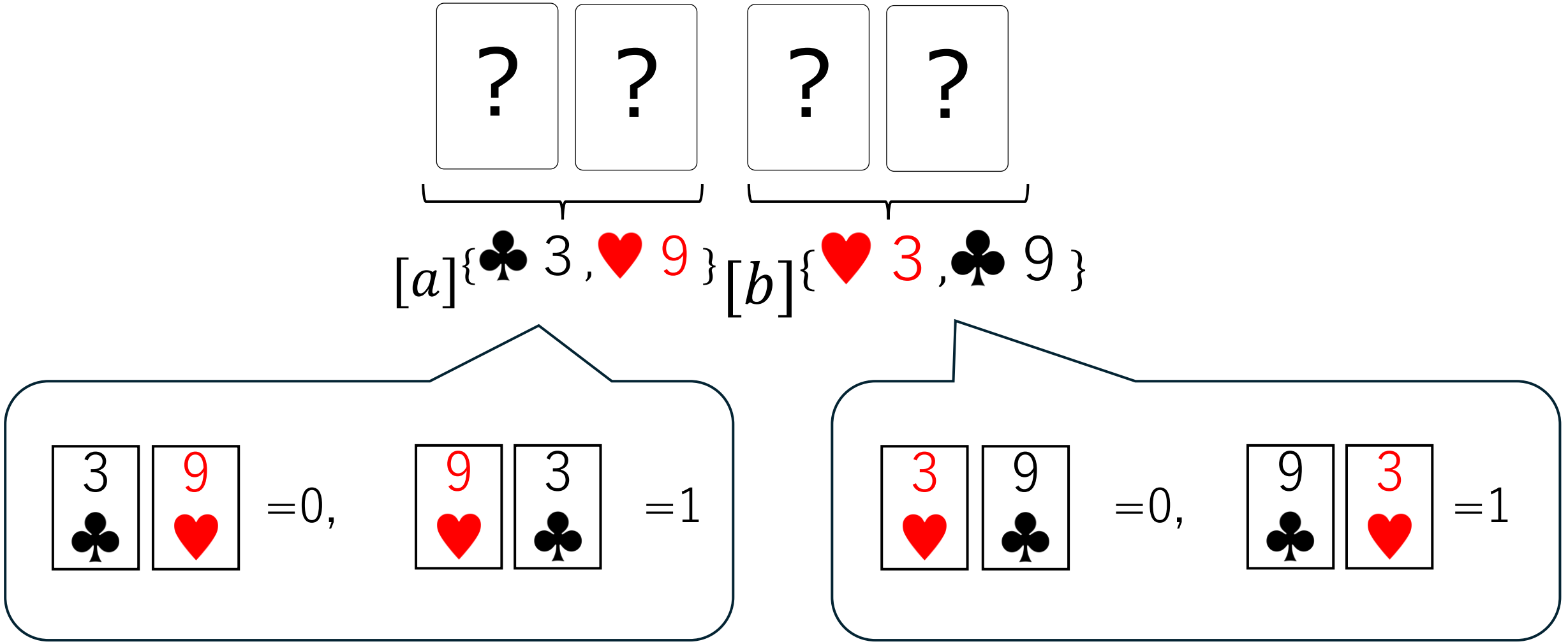
2入力非コミットANDプロトコル



$a \wedge b$

2種類のスートを2枚ずつ合計**4枚**を使用
ランダムカット:**1回** 半開示操作:**1回**

手順1:入力

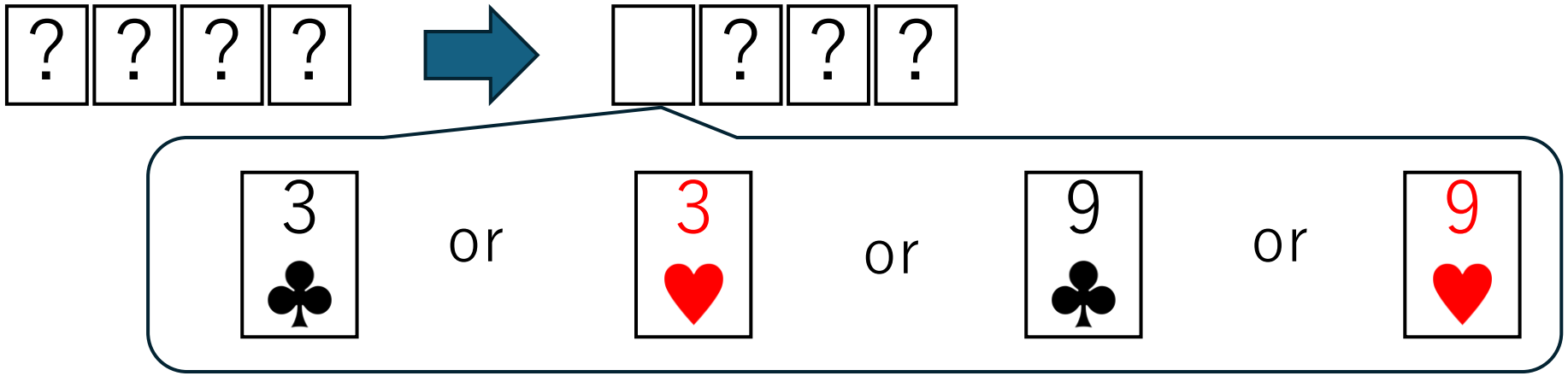


手順2:左端をめくる

1.ランダムカット(巡回的なシャッフル操作)を適用する

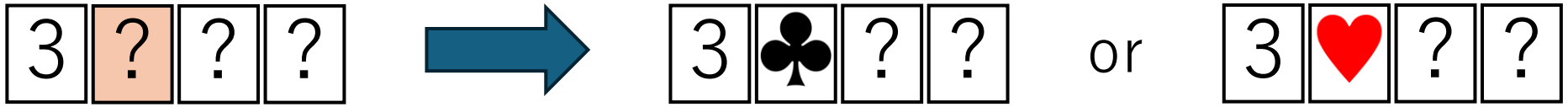
〈 ? ? ? ? 〉

2. 左端のカードをめくる

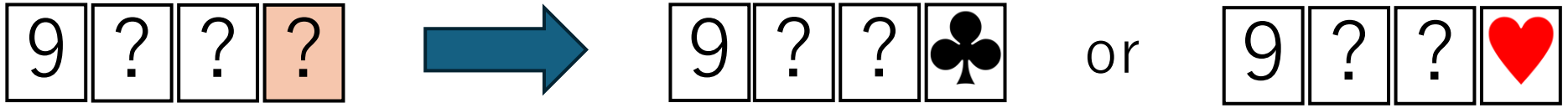


手順3:半開示操作

・左端の数字が3の場合、2枚目に半開示操作を適用する



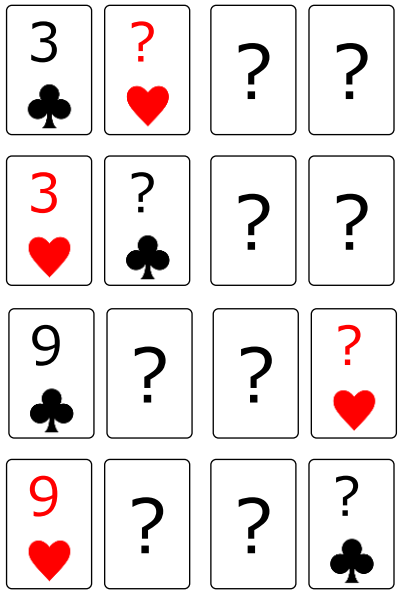
・左端の数字が9の場合、4枚目に半開示操作を適用する



出力

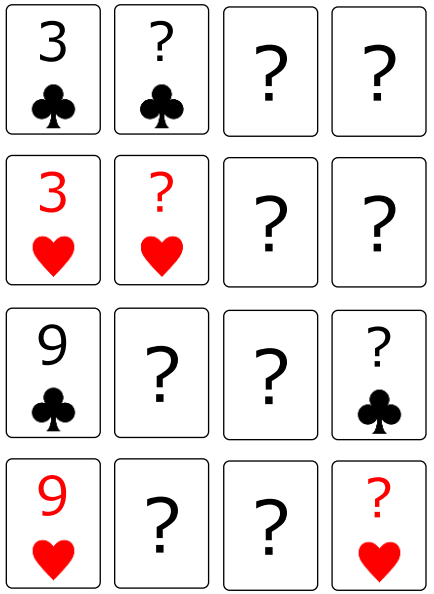
開示されているスートが

- 不一致 → 0
- 一致 → 1



a ∧ b=0

or



a ∧ b=1

安全性と正当性

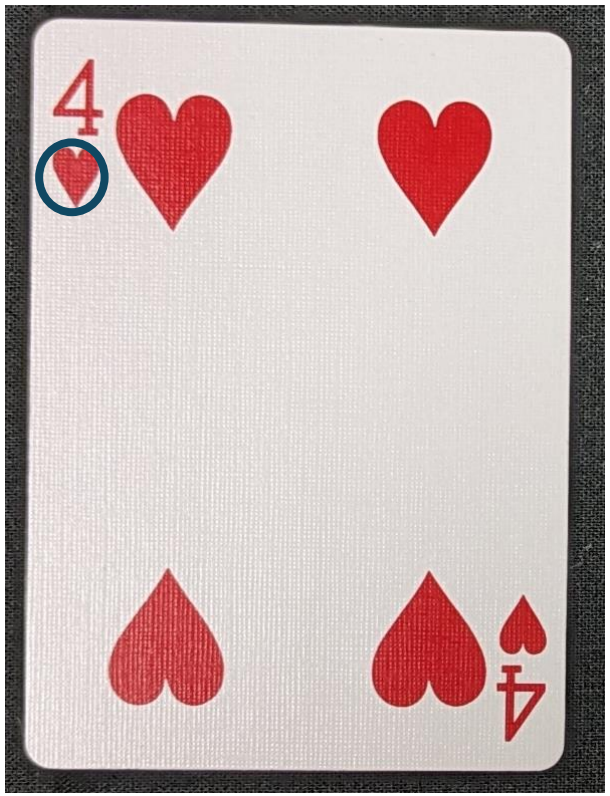
	入力カード列
(0,0)	3♣ 9♥ 3♥ 9♣
(0,1)	3♣ 9♥ 9♣ 3♥
(1,0)	9♥ 3♣ 3♥ 9♣
(1,1)	9♥ 3♣ 9♣ 3♥

めくる	♣3	♥3	♣9	♥9
出力	3♣ ?♥	3♥ ?♣	?♥ 9♣	?♣ 9♥
	3♣ ?♣	3♥ ?♥	?♣ 9♣	?♥ 9♥

部分開示操作

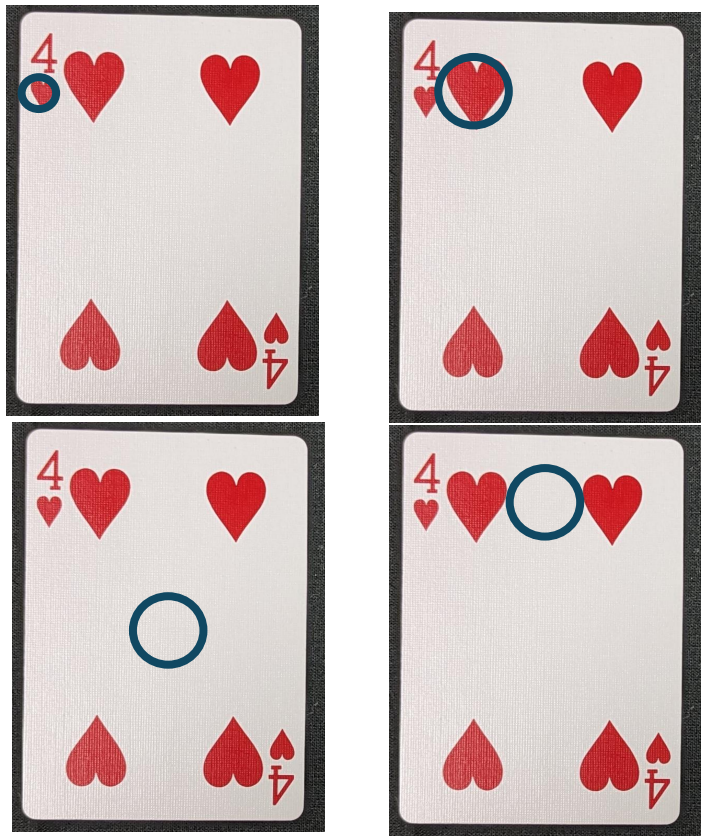
半開示操作の開示位置を一般化

半開示操作



左上のスートのみを開示

部分開示操作

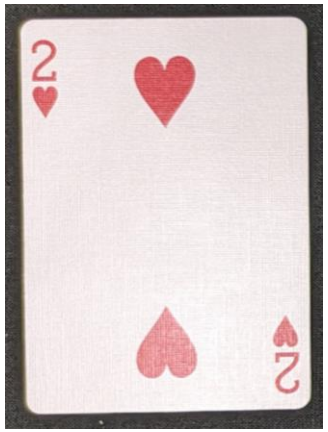


etc...

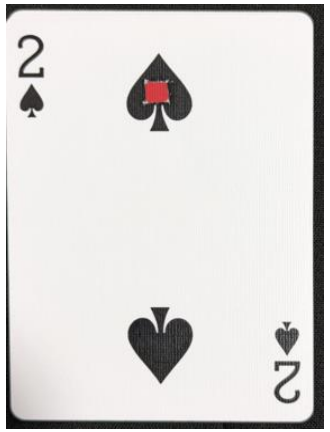
任意の部分を開示

部分開示操作

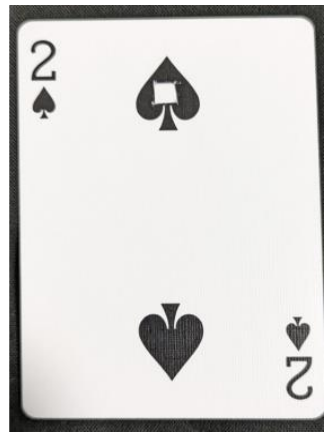
カードの任意の部分のみを開示する操作



穴を空けたカバー



スート
あり



スート
なし

部分開示操作の使用例

カードの特定

カードベースプロトコルの中には
カード列から目的のカードを見つけ出したいケースがある。

Ex.)5♥を含むカード列から5♥を取り除く。

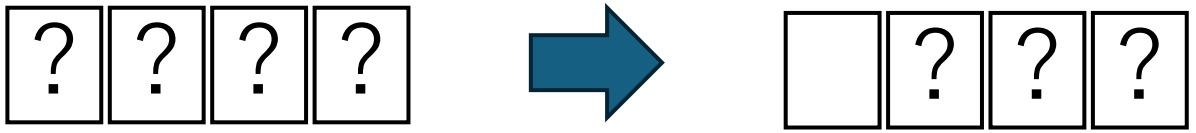
♥5を含むカード列　　♥5を含まないカード列

部分開示操作なしの特定プロトコル

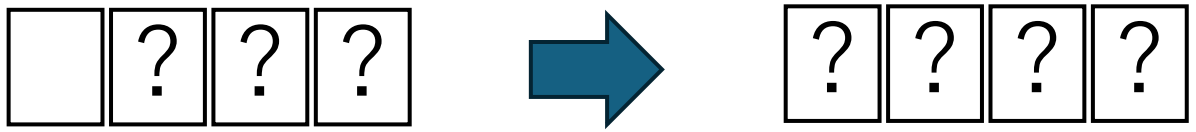
1.ランダムカット

〈 ? ? ? ? 〉

2.先頭のカードを開示して、目的のカードなら完了

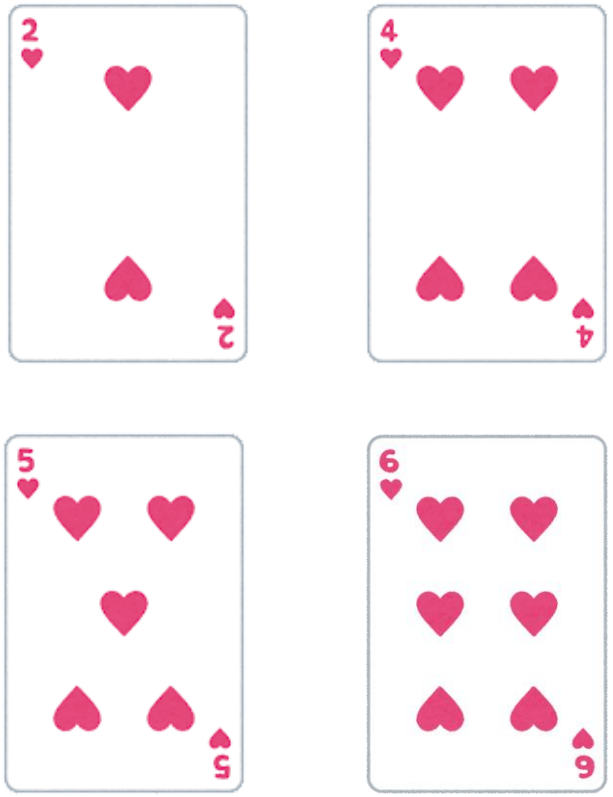


3.表になっているカードを裏向きにして手順1に戻る



部分開示操作を使用したカードの特定

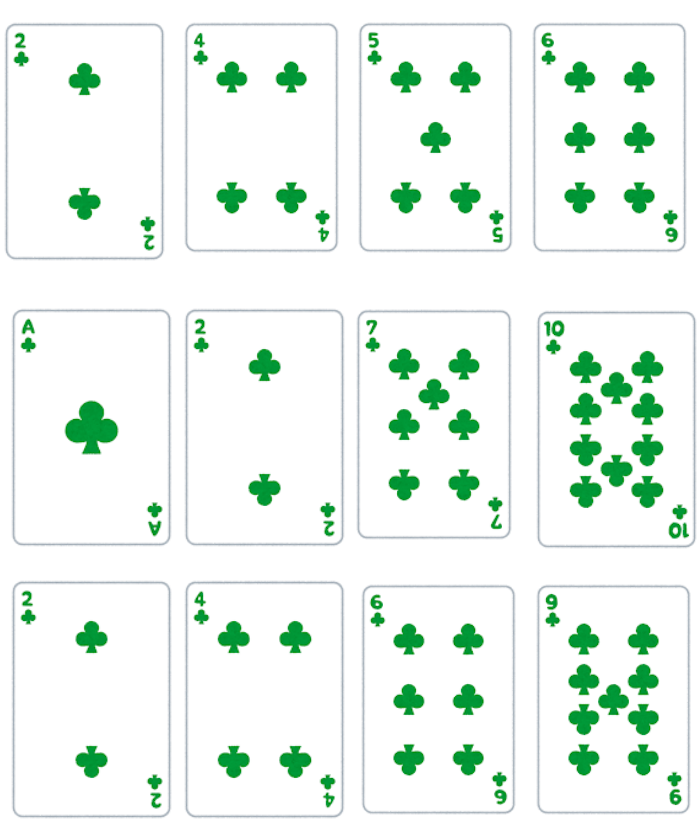
ランダムなカード列{2,4,5,6}からカード5を特定する

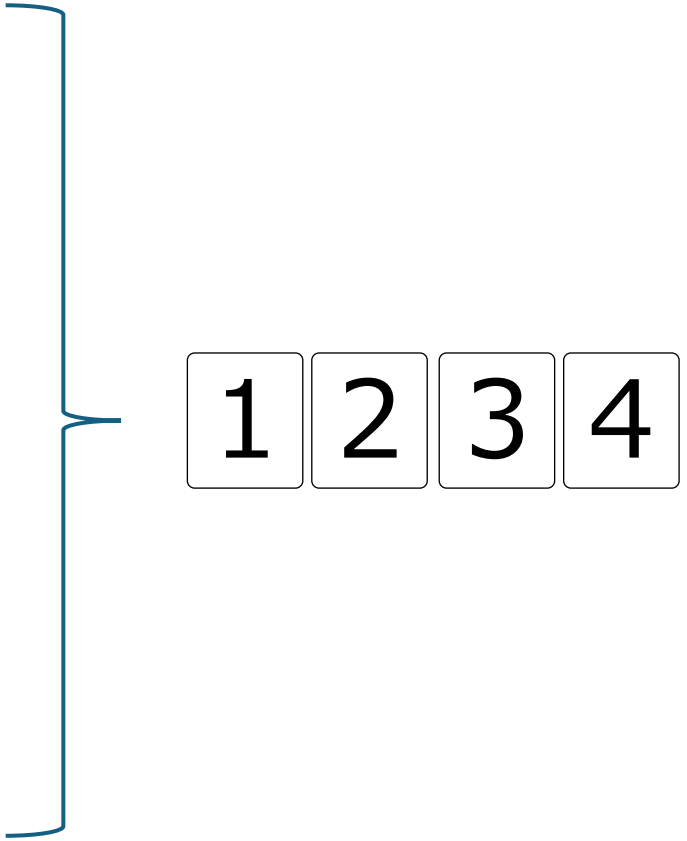


コミット型2入力ANDプロトコル

抽象カード

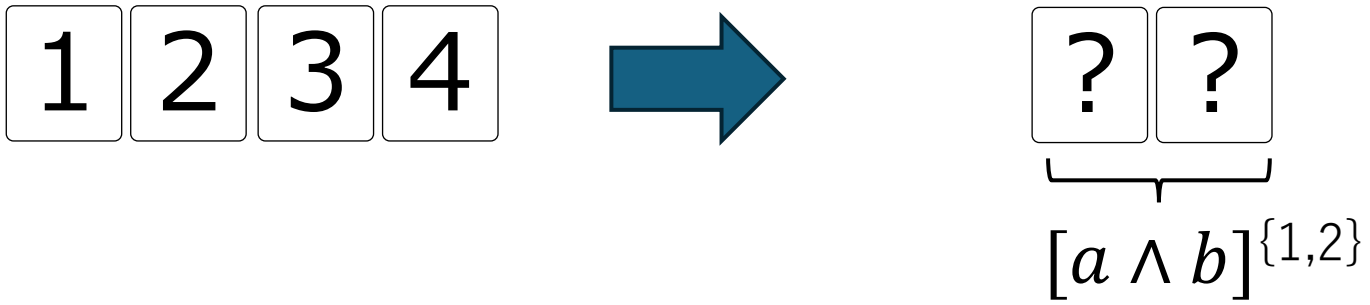
本プロトコルで使用するカードは抽象的な数字カードで表現する





プロトコルを実行可能なカード組の例

ANDプロトコルの概要

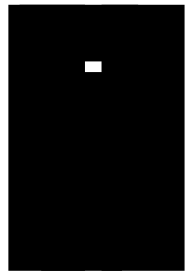
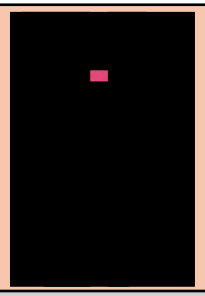
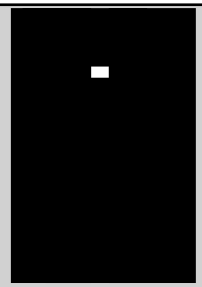
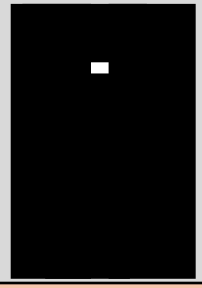
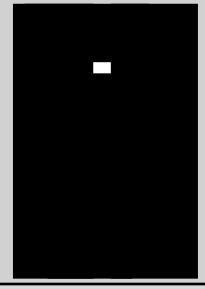
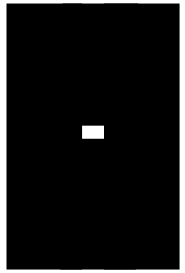
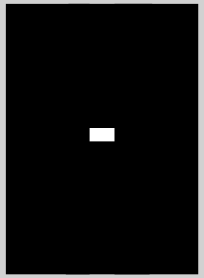
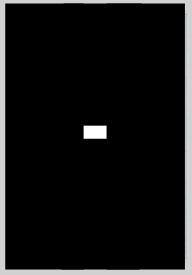
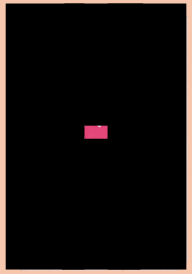
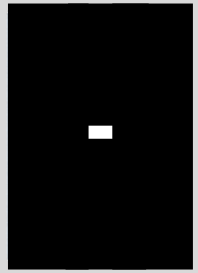
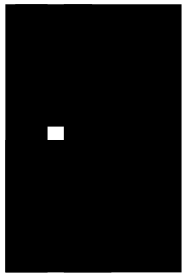
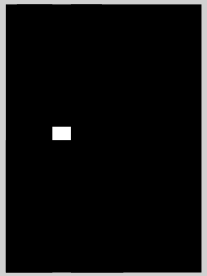
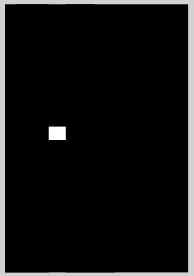
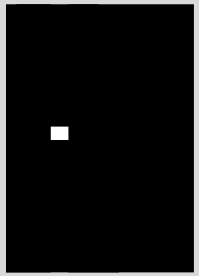
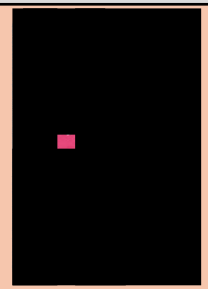


ランダムカット:3回，部分開示操作:8回

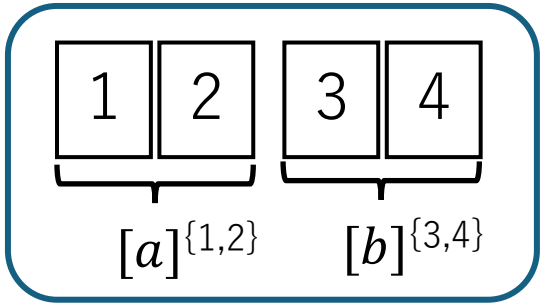
必要な部分開示操作は以下の3種類

- 1 のみスートが見える(= 1 を特定可能)
- 3 のみスートが見える
- 4 のみスートが見える

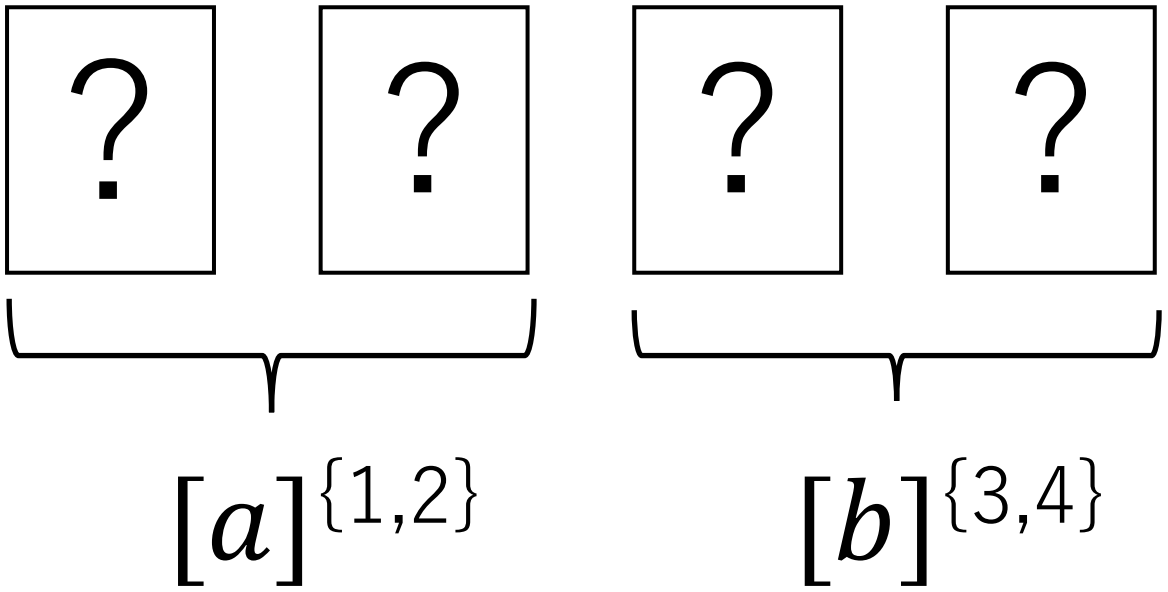
ANDプロトコルの条件を満たすカード組の例

カード / 開示位置	1	2	3	4
特定 1					
特定 3					
特定 4					

手順1 :入力

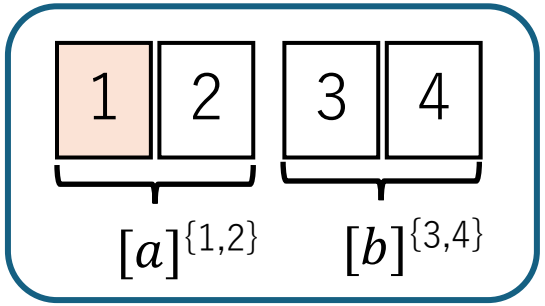


以下のように入力する



(a,b)	カード列
(0,0)	1 2 3 4
(0,1)	1 2 4 3
(1,0)	2 1 3 4
(1,1)	2 1 4 3

手順2: 1 を先頭にする



1. ランダムカットを適用する

〈 ? ? ? ? 〉

2. 部分開示操作で 1 を特定する

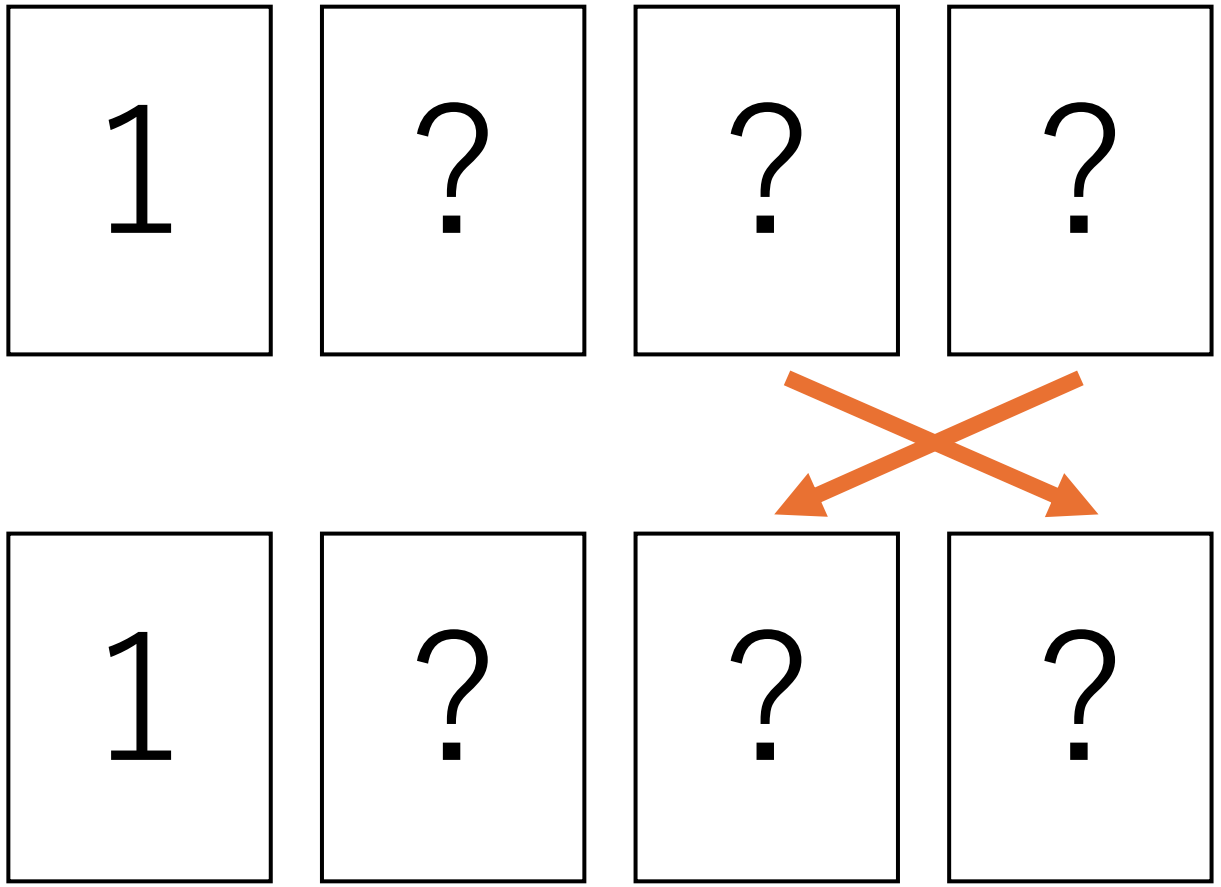
? ? ? ? ⇒ ? ? 1 ?

3. 巡回的に 1 を先頭にする

? ? 1 ? ⇒ 1 ? ? ?

(a,b)	カード列
(0,0)	**1** 2 3 4
(0,1)	**1** 2 4 3
(1,0)	2 **1** 3 4
(1,1)	2 **1** 4 3
(0,0)	**1** 2 3 4
(0,1)	**1** 2 4 3
(1,0)	**1** 3 4 2
(1,1)	**1** 4 3 2

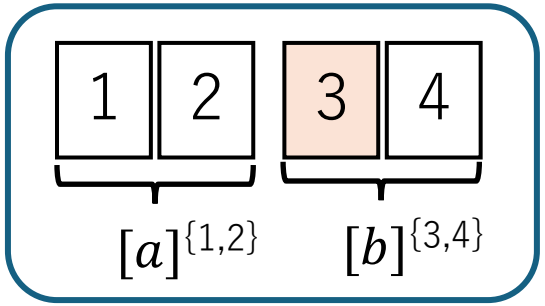
手順3:右側二枚を入れ替える



(a,b)	カード列
(0,0)	1 2 3 4
(0,1)	1 2 4 3
(1,0)	1 3 4 2
(1,1)	1 4 3 2

(0,0)	1 2 4 3
(0,1)	1 2 3 4
(1,0)	1 3 2 4
(1,1)	1 4 2 3

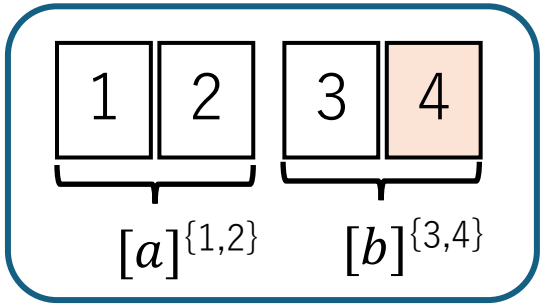
手順4: 3を除去する



1. ランダムカットを適用する
   〈 ? ? ? ? 〉
2. 部分開示操作で 3 を特定する
   ? ? ? ? ⇒ 3 ? ? ?
3. 3 をカード列から取り除く
   3 ? ? ? ⇒ □ ? ? ?

(a,b)	カード列
(0,0)	1 2 4 **3**
(0,1)	1 2 **3** 4
(1,0)	1 **3** 2 4
(1,1)	1 4 2 **3**
(0,0)	1 2 4
(0,1)	4 1 2
(1,0)	2 4 1
(1,1)	1 4 2

手順5:出力する



1. ランダムカットを適用する

〈 ? ? ? 〉

2. 部分開示操作で 4 を特定する

? ? ? → ? 4 ?

3. 巡回的に 4 を先頭にする

? 4 ? → 4 ? ?

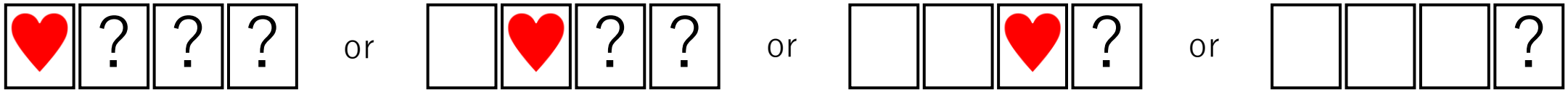
$[a \wedge b]^{\{1,2\}}$

(a,b)	カード列	
(0,0)	1 2 **4**	
(0,1)	**4** 1 2	
(1,0)	2 **4** 1	
(1,1)	1 **4** 2	
(0,0)	**4** 1 2	
(0,1)	**4** 1 2	$= 0$
(1,0)	**4** 1 2	
(1,1)	**4** 2 1	$= 1$

安全性

カードがめくられるのは手順2,4,5の部分開示操作のときのみ

手順2,4のランダムカットの後の部分開示操作結果は以下の4通り

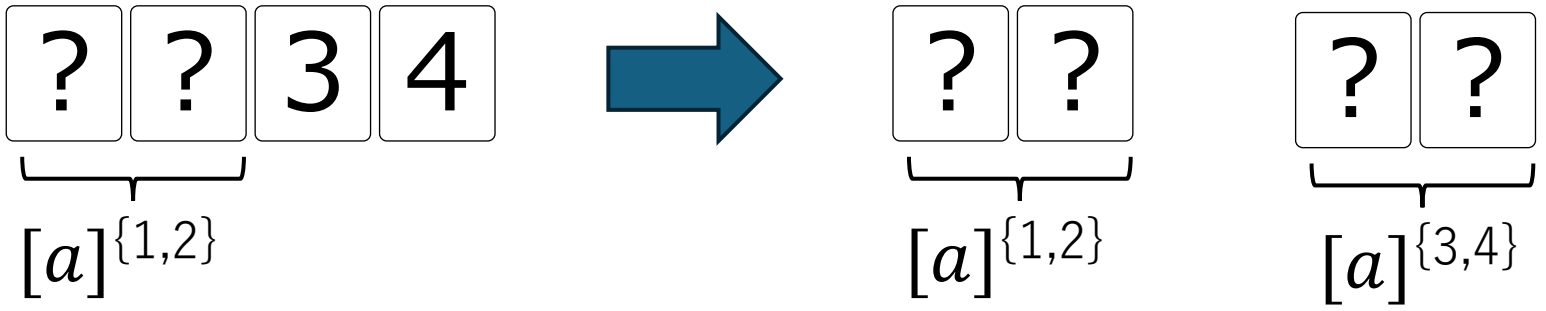


スートが見えたカード以外は区別がつかない(手順5も同様)

入力情報は洩れていない

4枚COPYプロトコル

COPYプロトコルの概要

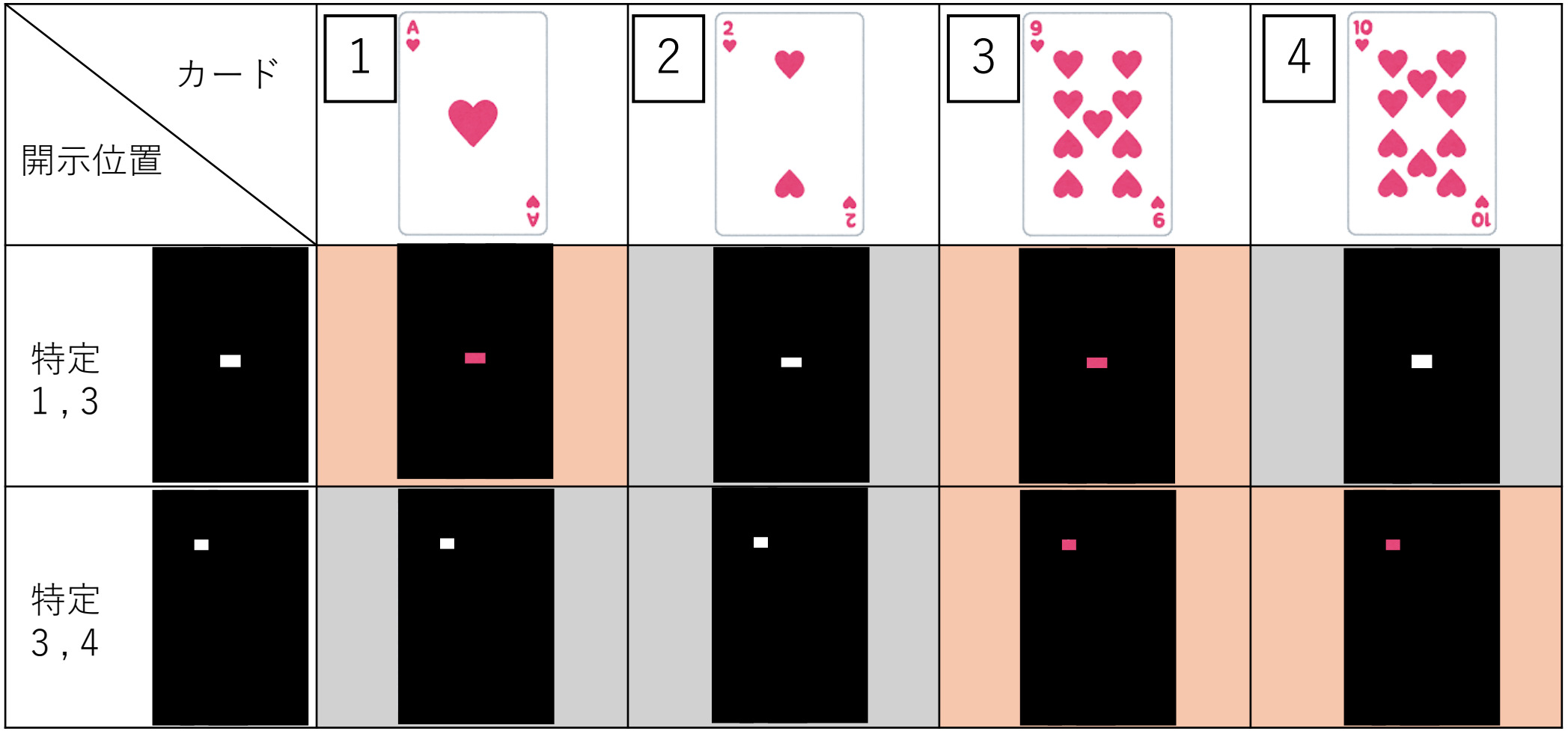


ランダムカット:3回，部分開示操作:4回

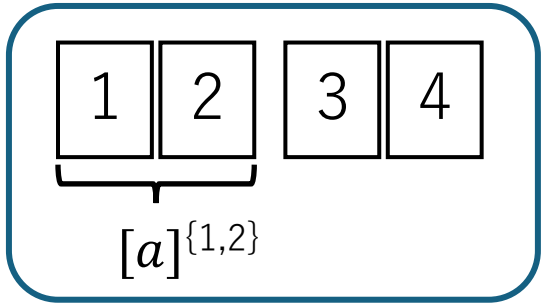
必要な部分開示操作は以下の2種類

- 1 と 3 のみスートが見える
- 3 と 4 のみスートが見える

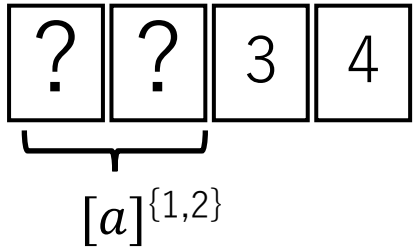
COPYプロトコルの条件を満たすカード組の例



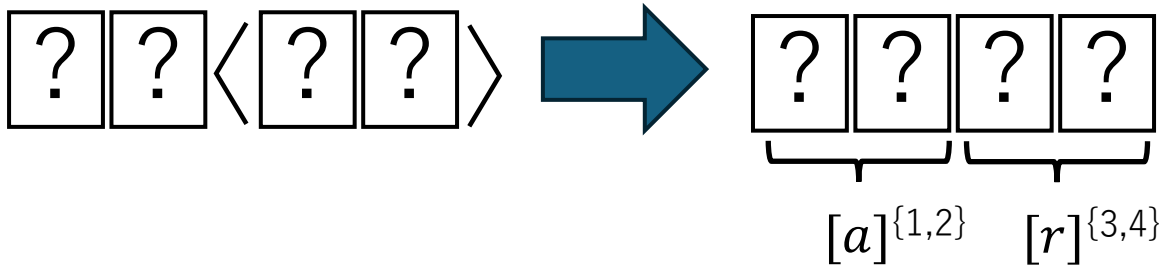
手順1:入力



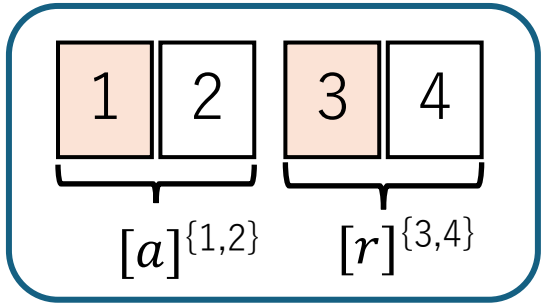
1. 以下のようにカードを並べる



2. 右側二枚にランダムカットを適用する

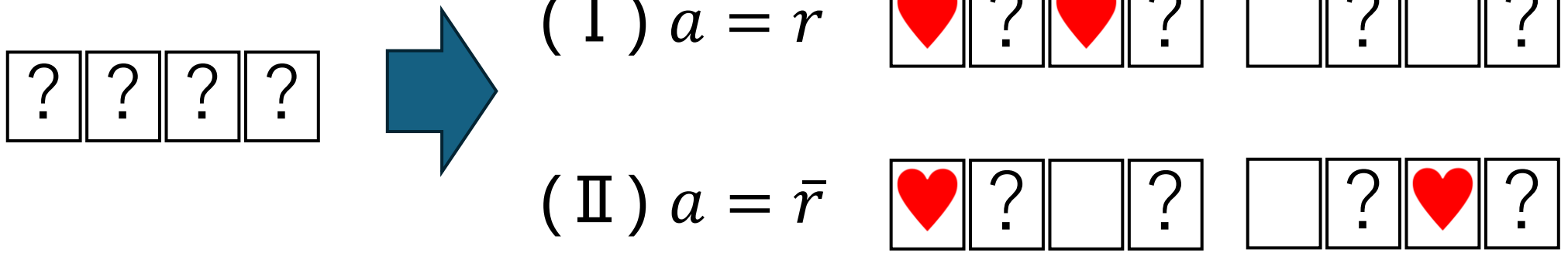


手順2:値の一致を確認

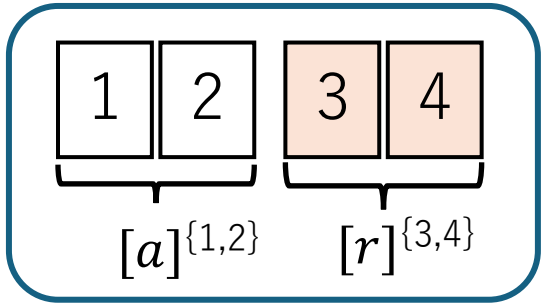


1. ランダムカットを適用する
   〈 ? ? ? ? 〉

2. 1枚目と3枚目に 1 または 3 を特定する部分開示操作を適用する



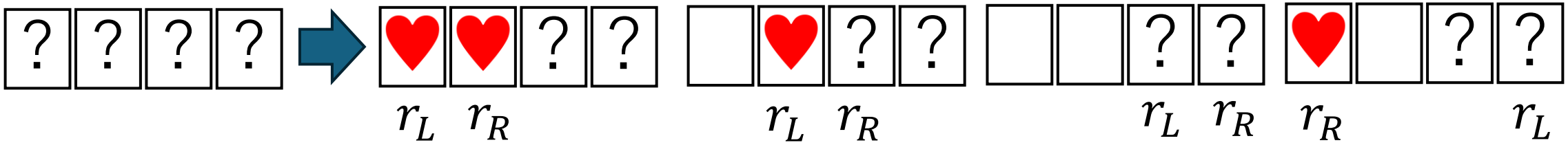
手順3:コミットメントを特定



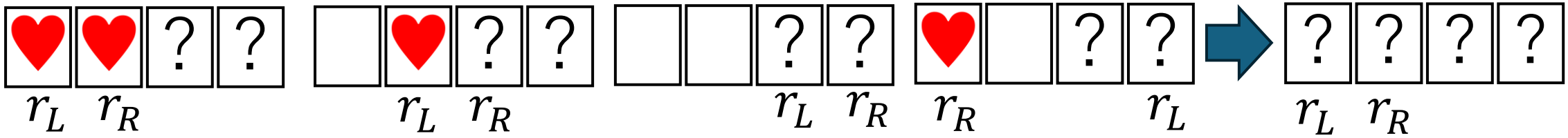
1. ランダムカットを適用する

〈 ? ? ? ? 〉

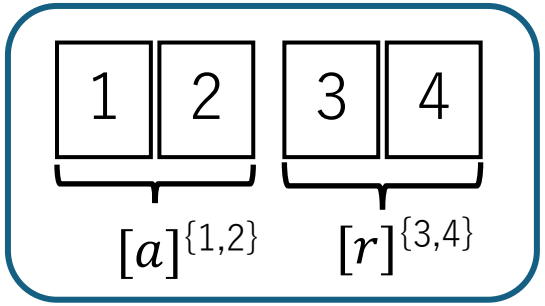
2. 左側2枚に対して 3 または 4 を特定する部分開示操作を適用する



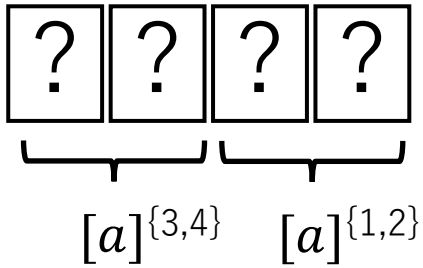
3. 巡回的にr_Lを先頭にする



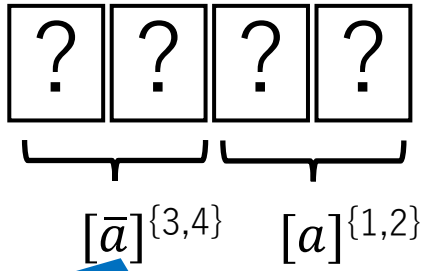
手順4:出力

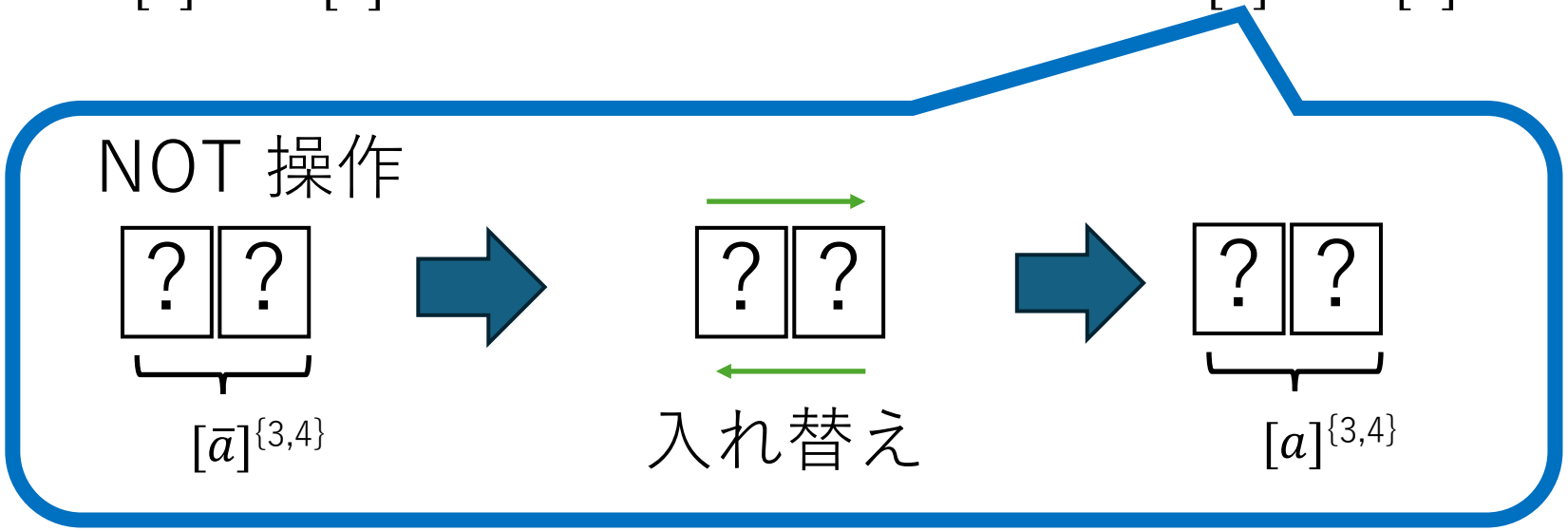


(Ⅰ) $a = r$の場合



(Ⅱ) $a = \bar{r}$の場合





まとめ

- 部分開示操作を提案した
  - 半開示操作の開示位置を一般化

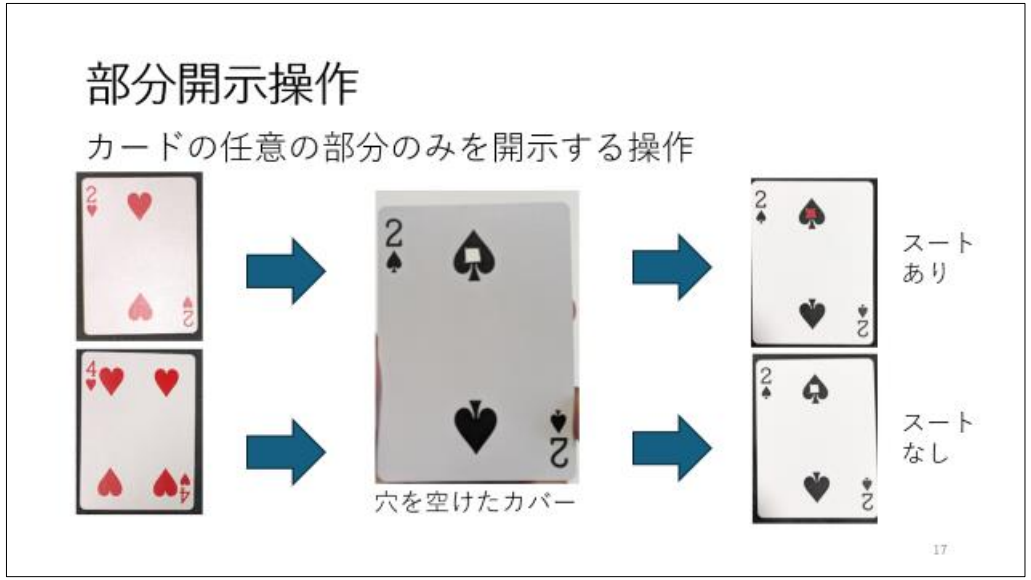
- 部分開示操作を用いて効率的なプロトコルを提案した
  - 4枚ANDプロトコル
  - 4枚COPYプロトコル

その後の研究の進展

- [本多-品川24]トランプ1デッキ26変数AND
- [本多-品川25]3入力/4入力非コミットAND

部分開示操作の詳細な話

Q.カバーはカード枚数に含まないの？



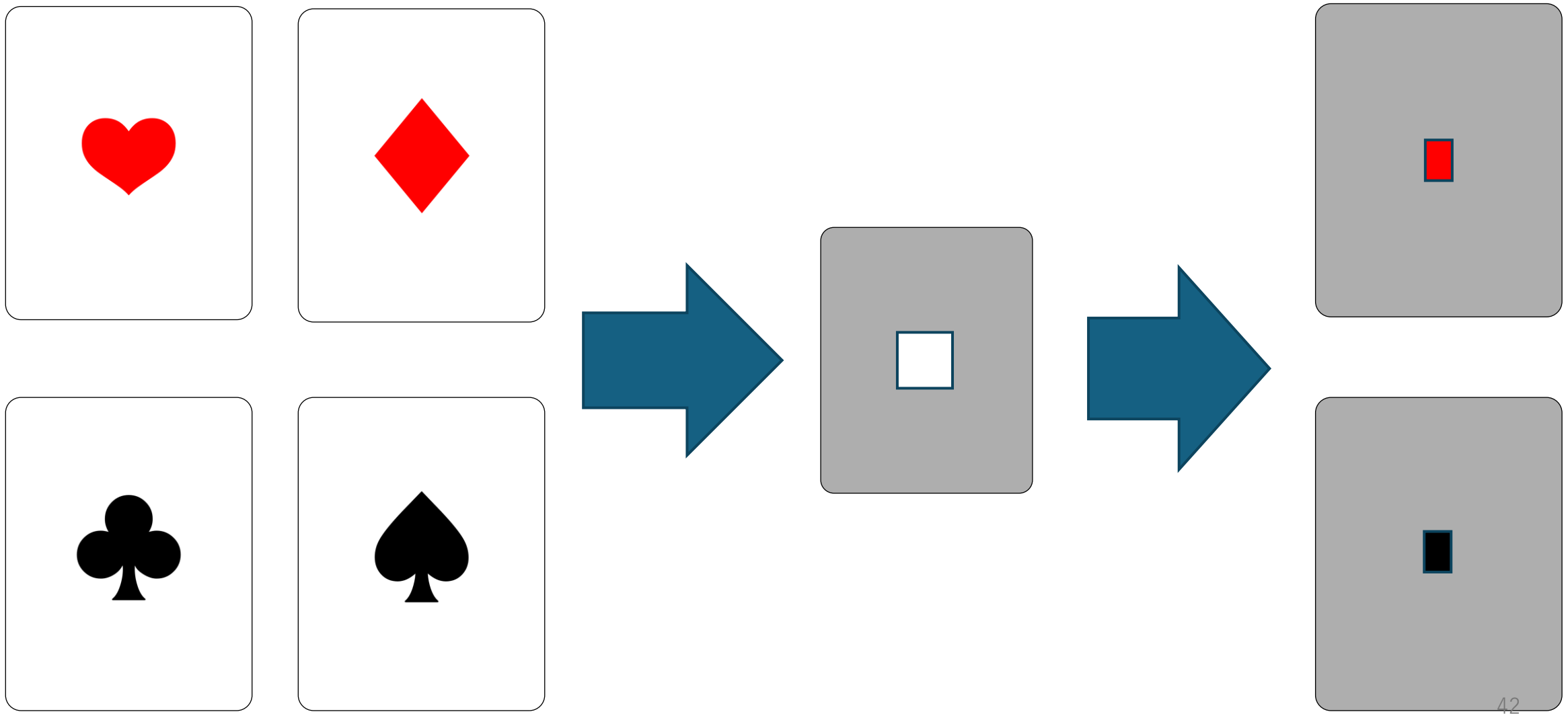
発表資料17ページ

A.トランプカードに穴を開けたものを、カバーとして使用する必要はないため、カード枚数には含まない

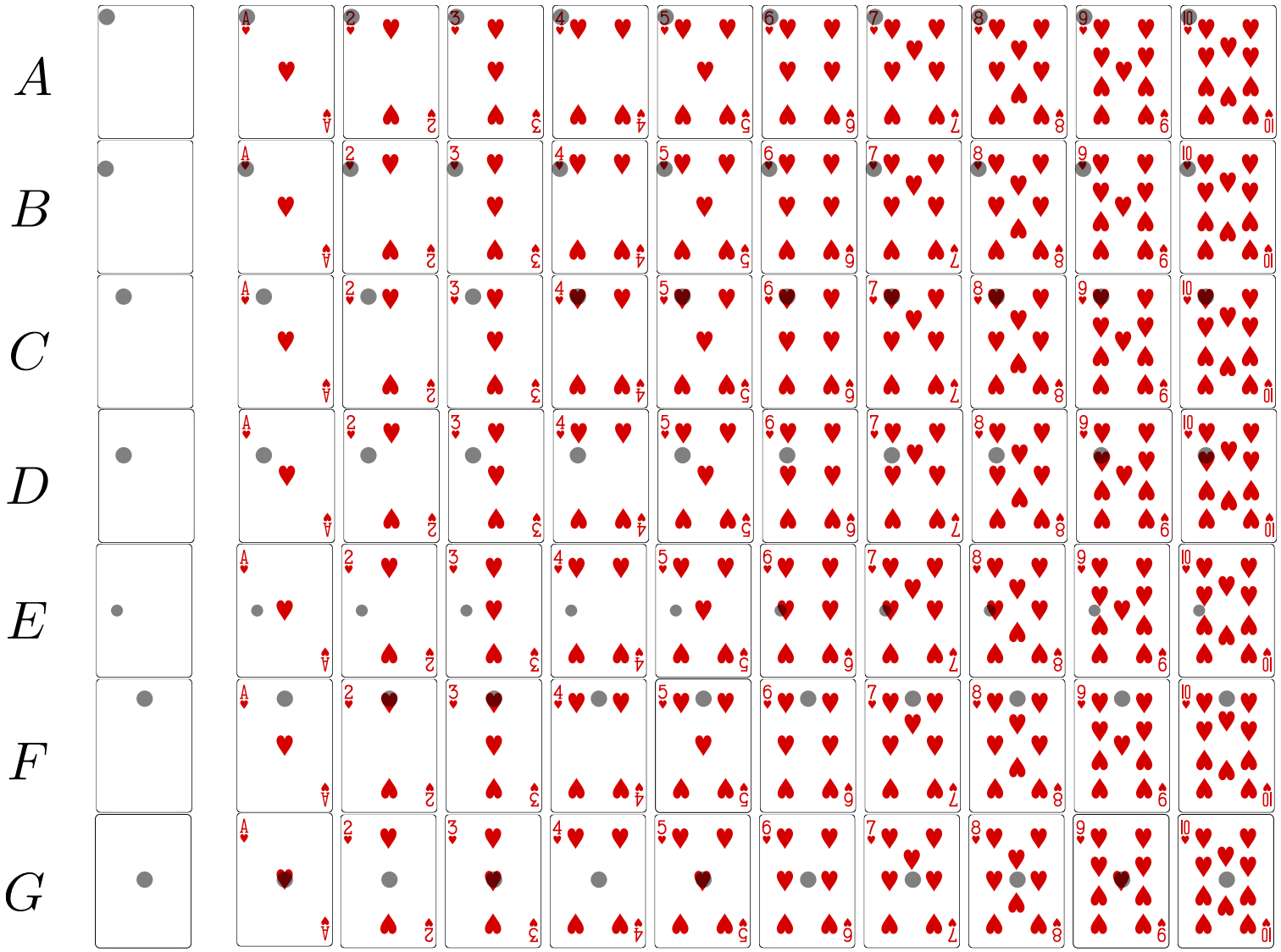
トランプカードをカバーにしている理由

- カードとサイズが同じで部分開示操作を適用しやすいから
- 表面のスートを利用すると、開示位置がわかりやすいから

トランプカード以外の部分開示操作



開示位置の例



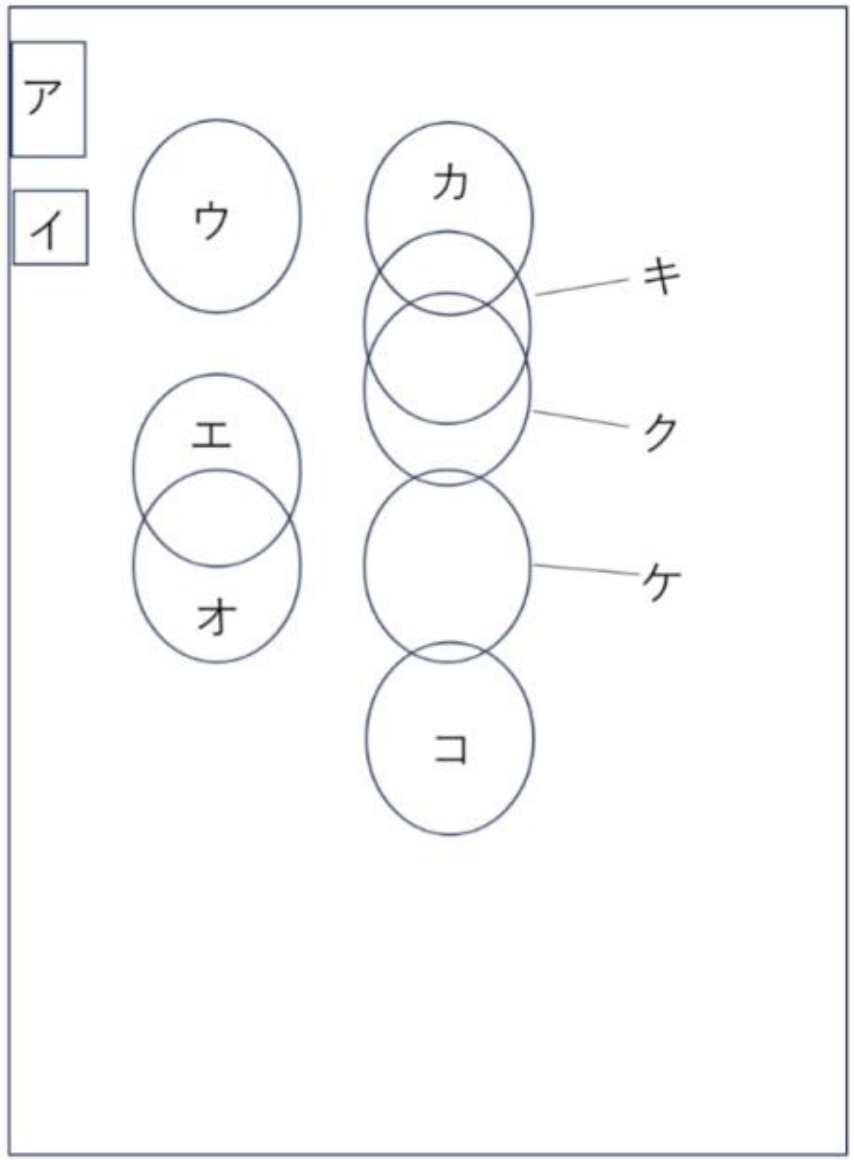


表 2: 各カードの開示位置と開示結果の関係

	1	2	3	4	5	6	7	8	9	10
ウ				○	○	○	○	○	○	○
エ									○	○
オ						○	○	○		
カ		○	○							
キ										○
ク							○	○		
ケ	○		○		○				○	
コ								○		