カードを用いた秘匿共通集合プロトコル

土井 アナスタシヤ[1,a)]　中井 雄士[1]　品川 和雅[2,3]　渡邉 洋平[1,3]　岩本 貢[1]

概要：トランプの様な物理的なカードを用いてマルチパーティ計算を実現する暗号技術をカードベース暗号と呼ぶ．カードベース暗号の分野では，これまでに論理演算プロトコルや多数決プロトコルなどの様々なプロトコルが提案されてきたが，秘匿集合計算 (Private Set Intersection: PSI) に関する研究は存在しない．PSI とは複数のパーティがそれぞれ保持する集合に関し，必要以上の情報を漏らすことなく共通集合演算の結果を算出するプロトコルであり，マルチパーティ計算における重要な研究テーマの一つである．本論文では，カードベース暗号において初めて PSI に焦点を当てその実現手法を示す．カードベース暗号には，すべての操作を公開することを仮定した操作モデルとプライベートな操作を許したモデルの 2 つの操作モデルがある．本論文ではそれぞれのモデルにおいて PSI プロトコルを提案する．まず，既存のカードベース AND プロトコルを用いることで，両操作モデル下で PSI プロトコルを実現できることを示す．その後，それぞれの操作モデルにおいて，これら AND プロトコルベースの PSI プロトコルを効率化する方式を提案する．

キーワード：マルチパーティ計算，カードベース暗号，秘匿共通集合計算

Card-based Cryptographic Protocols for Private Set Intersection

Anastasiia Doi[1,a)]　Takeshi Nakai[1]　Kazumasa Shinagawa[2,3]　Yohei Watanabe[1,3]
Mitsugu Iwamoto[1]

Abstract: Card-based cryptography is a cryptographic technique that realizes multiparty computation using physical cards. In card-based cryptography, protocols for various functions, e.g., logic operations and the majority voting, have been proposed. However, there is no research that focuses on Private Set Intersection (PSI). PSI is a cryptographic protocol that enables parties to compute the intersection of set of items while keeping the other items secret, and it is one of the most important research topics in multiparty computation. This paper focuses on PSI in card-based cryptography for the first time, and we propose four card-based PSI protocols. In card-based cryptography, there are two operation models: one assumes that all operations are performed publicly, and the other allows private operations. In this paper, we propose PSI protocols under each model. We first show that PSI can be realized under each model by utilizing the existing card-based AND protocols. Furthermore, we propose more efficient PSI protocols than the PSI protocols based on AND protocols under each model.

Keywords: Multi-party computation, Card-based cryptography, Private set intersection

1. はじめに

1.1 研究背景

マルチパーティ計算 (MPC) は，複数の参加者がそれぞれ持つ情報を秘匿したまま，それらの情報を入力値とした関数の計算を参加者同士で協調して行う暗号プロトコル

[1] 電気通信大学
The University of Electro-Communications
[2] 茨城大学
Ibaraki University
[3] 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology (AIST)
[a)] doi.ana@uec.ac.jp

である．一般的な MPC は，計算機への実装が想定されているが，計算機を用いずに物理的な道具を用いて構成される MPC も提案されている．その中で，本研究はトランプのような物理的なカードを用いて MPC を実現するカードベース暗号 [1] を扱う．

マルチパーティ計算における重要なテーマの一つに，秘匿共通集合プロトコル (PSI: Private Set Intersection)[2] がある．PSI とは, 全体集合 U に対して m 人のプレイヤがそれぞれ集合 $X_1, \ldots, X_m \subseteq U$ を保持し，これらを入力として積集合 $X_1 \cap \cdots \cap X_m$ を計算するが，$x \notin X_1 \cap \cdots \cap X_n$ に対しては，x がどの集合に属するかを一切情報をもらさないプロトコルである．カードベース暗号の研究は，AND 演算などの論理演算プロトコルから多数決プロトコルなど様々なプロトコルが扱われてきたが，PSI をテーマにしたカードベース暗号に関する研究はこれまでに存在しなかった．

1.2 本研究の貢献

本稿では，カードベース暗号における PSI プロトコルの実現手法を提案する．カードベース暗号には，すべて操作を公開することを仮定したシャッフルベースモデルと，プライベートな操作を許した秘匿置換ベースモデルの 2 つが存在する．本稿では，それぞれのモデルにおいて，PSI プロトコルを 2 つずつ提案する．まず一つ目の方式として，既存の AND プロトコルを用いて実現できるプロトコルをそれぞれのモデルに対して提案する．次に，二つ目の方式として，これら 2 つの AND ベースのプロトコルをカード枚数や手順数の観点から効率化したプロトコルを各モデルに対して提案する．

提案方式の効率性を表 1 および表 2 にまとめた (n を全体集合のサイズとする)．表 1 ではシャッフルシャッフルベースの PSI プロトコルを比較した．プロトコル 1 はシャッフル技術の中で最初に考案された 5-card AND プロトコル [1] をベースに構成した PSI プロトコルである．プロトコル 2 は，パイルスクランブルシャッフルと呼ばれる [3] シャッフルを用いて構成した方式である．ただし，プロトコル 2 ではプロトコル 1 とは異なり，プレイヤの入力値の集合のサイズの情報が漏洩することを許容する．表 1 より，プロトコル 2 はカード枚数，シャッフル数の両評価項目において効率化に成功していることがわかる．

また，表 2 は秘匿置換ベースの PSI プロトコルの比較表である．プロトコル 3 は 3-card AND プロトコル [4] をベースに構成した PSI プロトコルである．プロトコル 4 は，プロトコルへの逐次比較のアイデアを導入したプロトコルである．本方式は，秘匿置換の回数および通信回数がプロトコル 3 よりも増えているが，カード枚数の観点では効率化に成功している．

表 1 シャッフルベースの 2 者間 PSI プロトコルの比較

プロトコル	カード枚数	シャッフル回数
プロトコル 1	$5n$	n
プロトコル 2	$3n$	1

表 2 秘匿置換ベースの 2 者間 PSI プロトコルの比較

プロトコル	カード枚数	秘匿置換回数	通信回数
プロトコル 3	$3n$	1	1
プロトコル 4	$2n+2$	n	$n+1$

2. 準備

2.1 カードベース暗号

本稿では表面が $\boxed{\clubsuit}\boxed{\heartsuit}$ の 2 種類のカードを用いることとする． 全てのカードの裏面は $\boxed{?}$ で表し，区別できないものとする．

2 値 2 枚のカードで $0 \mapsto \boxed{\clubsuit}\boxed{\heartsuit}$, $1 \mapsto \boxed{\heartsuit}\boxed{\clubsuit}$ と表現し，$\boxed{?}\boxed{?} \in \{0,1\}$ とした裏面のカード組をコミットメントと呼ぶ．$0 \mapsto \boxed{\clubsuit}$, $1 \mapsto \boxed{\heartsuit}$ に従い，$\boxed{?} \in \{0,1\}$ とした裏面のカードを 1 枚表現コミットメントと呼ぶ．

次節では，シャッフルベースモデルと秘匿置換ベースモデルについて，それぞれの操作モデルを説明する．

2.1.1 シャッフルベースモデル

カードベース暗号における従来研究の多くは，すべての操作をテーブル上などの公開の場で行うシャッフルベースモデルを仮定している．シャッフルベースモデルでは次の 3 つの操作を用いてプロトコルを構成する．

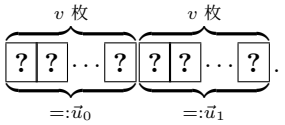
- 公開置換：確定的な置換操作
- 反転：カードの裏表を公開の場で変える操作
- シャッフル：確率的な置換操作

シャッフルベースモデルにおけるプロトコルの効率性はカード枚数，シャッフル回数にて評価する．

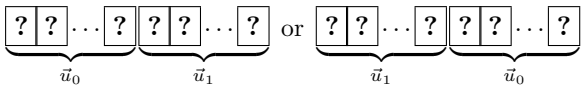
シャッフルはすべての操作を公開するとした仮定の下で，プライバシーを保証するために特に重要な操作である．本稿で用いるシャッフルについて以下で説明する．

ランダム二等分割カット [5]:

ランダム二等分割カットとは，偶数枚のカードを左右同枚数に分け，左右をランダムに入れ替えるシャッフルである．つまり，整数値 v に対し，$2v$ 枚のカードがあるとする．これを左右同じ枚数に分ける（それぞれ $\vec{u}_0, \vec{u}_1$ とする）．



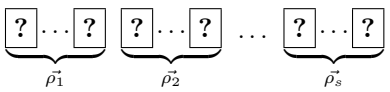
この $2v$ 枚のカード組にランダム二等分割カットを適用した結果として，



2 つの並びがそれぞれ 1/2 の確率で発生する．つまり，結果の並びを $(\vec{u}_r, \vec{u}_{1-r})$ としたとき，一様ランダムに生成された乱数 $r \in \{0,1\}$ はすべてのプレイヤが特定できない．

パイルスクランブルシャッフル [3]:

パイルスクランブルシャッフルとはカード束の列に対するシャッフルを指す．ある正の整数 s に対し，s 個のカード束が $\vec{\rho_i}$，カード束の列が $(\vec{\rho_1}, \vec{\rho_2}, \ldots, \vec{\rho_s})$ で与えられているとする．



なお，各カード束は同枚数で構成されているとする．r を s 次対称群から一様ランダムに選ばれた置換であるとしたとき，パイルスクランブルシャッフルは結果として

$$(\vec{\rho}_{r^{-1}(1)}, \vec{\rho}_{r^{-1}(2)}, \cdots, \vec{\rho}_{r^{-1}(s)})$$

を得るシャッフルである．このとき，一様ランダムに選ばれた置換 r はすべてのプレイヤが特定できない．

2.1.2 秘匿置換ベースモデル

秘匿置換ベースモデルは Marcedone ら [4] および Nakai ら [6] が提案した操作モデルである．秘匿置換ベースモデルでは，シャッフルベースモデルとは異なり，プレイヤがカードを背に隠すなどして秘匿した状態でカードを並べ替える操作を許す．秘匿置換ベースモデルでは次の 4 つの操作を用いてプロトコルを構成する．

- 公開置換：公開された場で行う置換操作
- 秘匿置換：プライベートな場で行う置換操作
- 反転：カードの裏表を公開の場で変える操作
- 通信：カードを相手に渡す操作

この操作モデルでは，シャッフルは一つの操作ではなく，2 つの秘匿置換と 1 つの通信により実現される操作であると実現することができる．例えば，ランダム二等分割カットは，2 人のプレイヤ Alice と Bob によって以下のように実現されると解釈できる．手順の開始時には，$2v$ 枚のカードは Alice が所持しているとする．

(1) Alice は $r_A \in \{0,1\}$ を確率 1/2 で決定し，秘匿置換により $(\vec{\rho}_0, \vec{\rho}_1)$ を r_A 回入れ替える．

(2) Alice は，操作後のカード組 $(\vec{\rho}_{r_A}, \vec{\rho}_{1-r_A})$ を Bob へ手渡す．

(3) Bob は，$r_B \in \{0,1\}$ を確率 1/2 で決定し，秘匿置換により $(\vec{\rho}_{r_A}, \vec{\rho}_{1-r_A})$ を r_B 回入れ替える．

結果のカード組の並びは，$(\vec{\rho}_{r_A \oplus r_B}, \vec{\rho}_{r_A \oplus r_B})$ となる ($\oplus$ は排他的論理和を表す)．このとき，それぞれのプレイヤが生成した乱数は秘匿されているため，結果としてカード組が何回入れ替えられたのかはすべてのプレイヤが特定できず，ランダム二等分割カットと同様の効果が得られる．

秘匿置換ベースモデルではカード枚数や秘匿置換回数・通信回数でプロトコルの効率性を評価する．

2.2 既存のカードベース AND プロトコル

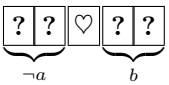
AND プロトコルとは，2 人のプレイヤ (Alice と Bob とする）がそれぞれ保持するブール値 a, b を入力値として，必要以上の情報を漏らすことなく $a \wedge b$ を計算するプトロコルである．

2.2.1 シャッフルベースの AND プロトコル

5-card AND プロトコル [1]：den Boer により提案された 5 枚のカードと 1 回のシャッフルで AND プロトコルを実現する 5-card AND プロトコルを紹介する．以下にこのプロトコルの手順を示す．

手順の開始時，Alice と Bob はそれぞれ，a, b を表現するコミットメントを入力値として保持しているとする．また，追加のカードとして $\boxed{\heartsuit}$ を用意する．

(1) 以下の並びを作成する．



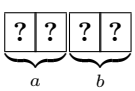
(2) 真ん中の $\boxed{\heartsuit}$ を裏にしたのち，5 枚のカードに対してランダムカット (巡回的なシャッフル) を適用する．

(3) 全てのカードを表にし，$\boxed{\heartsuit}$ が巡回的に 3 つ連続していたら $a \wedge b = 1$，そうでないならら $a \wedge b = 1$ が出力値となる．

4-card AND プロトコル [7]：Mizuki らが提案した 4 枚のカードと 2 回のシャッフルで AND プロトコルを実現する 4-card AND プロトコルを紹介する．以下に手順を示す．

手順の開始時，Alice と Bob はそれぞれ，a, b を表現するコミットメントを入力値として保持しているとする．

(1) 以下の並びを作成する．



(2) 4 枚のカード組に対して，ランダム二等分割カットを適用する．

(3) 真ん中 2 枚のカードにランダムカットを施す．

(4) 左から 2 枚目のカードを表にする．

- $\boxed{\clubsuit}$ であれば，新たに左から 4 枚目のカードを表にする．そのカードが $\boxed{\heartsuit}$ ならば $a \wedge b = 0$ であり，$\boxed{\clubsuit}$ ならば $a \wedge b = 1$ となる．
- $\boxed{\heartsuit}$ であれば，新たに左から 1 枚目のカードを表にする．そのカードが $\boxed{\clubsuit}$ ならば $a \wedge b = 0$ であり，$\boxed{\heartsuit}$ ならば $a \wedge b = 1$ となる．

2.2.2 秘匿置換ベースの AND プロトコル

本節では，Marcedone ら [4] が提案した 3-card AND プロトコルを紹介する. 以下にその手順を示す.

プロトコル開始時，Alice は $\boxed{\heartsuit}$ と $\boxed{\clubsuit}$ を各 1 枚ずつ, Bob は $\boxed{\clubsuit}$ を 1 枚ずつ保有する．

(1) Alice は $a = 0$ ならば $\boxed{\clubsuit}$ を, $a = 1$ ならば $\boxed{\heartsuit}$ を裏にして Bob に渡す.

(2) Bob は秘匿置換を用いて以下の操作を行う．

- $b = 0$ ならば，受け取ったカードの左に $\boxed{\clubsuit}$ を置く．
- $b = 1$ ならば，受け取ったカードの右に $\boxed{\clubsuit}$ を置く．

(3) Bob は左のカードを表に開く．

- $\boxed{\clubsuit}$ ならば，$a \wedge b = 0$ である．
- $\boxed{\heartsuit}$ ならば，$a \wedge b = 1$ である．

3. シャッフルベースの PSI プロトコル

3.1 AND プロトコルを用いた方式

本節では，AND プロトコルを用いて PSI プロトコルを実現する方法を提案する．

全体集合を $U = \{u_1, \ldots, u_n\}$ とする．$i = 1, \ldots, n$ に関して，$u_i \notin A$ ならば $a_i = 0$，それ以外ならば $a_i = 1$ とし，$a = (a_1, \ldots, a_n)$ とすることで，Alice の入力値の集合 A を 2 進数 a で表現できる (B も同様である)．以降，PSI プロトコルへの入力値を $a = (a_1, \ldots, a_n)$ のように表現した際は，この手順で集合を 2 進数へ変換した値であるとする．

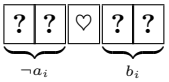
入力値の集合を 2 進数で表現した各ビットに対し，AND プロトコルを適用することで，PSI の結果 $A \cap B$ を 2 進数として得ることができる．プロトコル 1 に，5-card AND プロトコルを用いる際のシャッフルベースの PSI プロトコルの構成を示す.

プロトコル 1 5-card AND プロトコルをベースとした方式

Inputs：Alice: $a = (a_1, \ldots, a_n)$, Bob: $b = (b_1, \ldots, b_n)$

Setup：Alice と Bob はそれぞれ $a = (a_1, \ldots, a_n)$, $b = (b_1, \ldots, b_n)$ を表現するコミットメントの組を入力値として保持しているとする．追加のカードとして $\boxed{\heartsuit}$ を n 枚用意する．

(1) $i = 1, \ldots, n$ に関して，以下の並びを作成する．



(2) n 組ある 5 枚のカード列に対して，それぞれランダムカットを適用する．

(3) 全てのカードを表にする．このとき，$i = 1, \ldots, n$ に関する $a_i \wedge b_i$ の結果を得ることができ，$a_i \wedge b_i = 1$ となる u_i すべてからなる集合が所望の共通集合 $A \cap B$ である．

上記のプロトコルが必要とするカード枚数は $10n$ 枚である．また，シャッフル回数は n 回である．また，AND プロトコルとして，4-card AND プロトコルを採用した場合は，同様の手順で $8n$ 枚のカードと $2n$ 回のシャッフルで PSI プロトコルを実現できる．

Correctness：$a_i = 1$ かつ $b_i = 1$ (つまり，$a_i \wedge b_i = 1$) であるとは，$u_i \in A$ かつ $u_i \in B$ であるため，$u_i \in A \cap B$ である．一方で，$a_i \neq 1$ または $b_i \neq 1$ (つまり，$a_i \wedge b_i = 0$) であるとは，$u_i \notin A$ または $u_i \notin B$ であるため，$u_i \notin A \cap B$ である．従って，$a_i \wedge b_i = 1$ となる u_i すべてからなる集合が $A \cap B$ である．

Security：$a_i \wedge b_i = 0$ となるような i について，AND プロトコルの性質より，$(a_i, b_i) = (0,0), (0,1), (1,0)$ は区別できない．従って，Alice は $a_i = 0$ であるとき，$b_i = 1$ であるか特定できない．つまり，Alice は $u_i \notin A$ であるとき，$u_i \in B$ であるかどうか特定できない (Bob の場合も同様である)．従って，Alice，Bob はプロトコルから入出力以上の情報を得ることはできない．

3.2 パイルスクランブルシャッフルに基づく方式

3.1 節の AND プロトコルを用いて PSI プロトコルを実現する手法は，シャッフル回数が全体集合 U のサイズ n に比例してしまう．本節では，$3n$ 枚のカードとたった 1 回のシャフルで PSI プロトコルを実現できることを示す．ただし，入力値の集合のサイズ $|A|$ および $|B|$ の漏洩を許容する．プロトコル 2 にその手順を示す.

Correctness：手順 (3) で a'_i と b'_i のカード両方が $\boxed{\heartsuit}$ であるとは，$a'_i = 1$ かつ $b'_i = 1$ であり，それ以外の場合は $a'_i \neq 1$ または $b'_i \neq 1$ であることに注意する．$a'_i = 1$ かつ $b'_i = 1$ とは $u'_i \in A$ かつ $u'_i \in B$ である．従って，a'_i と b'_i の両方が $\boxed{\heartsuit}$ であるような u'_i すべてからなる集合が $A \cap B$ である．

Security：手順 (3) で，$i = 1, \ldots, n$ について a'_i, b'_i を表に

プロトコル 2 パイルスクランブルシャッフルに基づく方式

Inputs：Alice: $A = (a_1, \ldots, a_n)$, Bob: $B = (b_1, \ldots, b_n)$. ただし，$|A| = n_a$, $|B| = n_b$ とする．

Setup：Alice は，$\boxed{\clubsuit}$ を $n - n_a$ 枚，$\boxed{\heartsuit}$ を各 n_a 枚所持する．Bob は，$\boxed{\clubsuit}$ を $n - n_b$ 枚，$\boxed{\heartsuit}$ を各 n_b 枚所持する．それ以外に，全体集合 U の各要素が表面に描かれた n 枚のカード組 $(u_1, \ldots, u_n)$ を用意する．

(1) $i = 1, \ldots, n$ について，$\vec{\rho_i} = (u_i, a_i, b_i)$ とする．

(2) $(\vec{\rho_1}, \ldots, \vec{\rho_n})$ に対して，パイルスクランブルシャッフルを適用する．適用後のカード組を $(\vec{\rho'_1}, \ldots, \vec{\rho'_n})$ とする． ただし，$i = 1, \ldots, n$ について，$\vec{\rho'_i} = (u'_i, a'_i, b'_i)$ とする．

(3) $i = 1, \ldots, n$ について，a'_i, b'_i を表に開く．このとき，開いたカードの両方が $\boxed{\heartsuit}$ であるならば，u'_i も合わせて表に開く

(4) $(u_1, \ldots, u_n)$ の内，表に開かれたカードに描かれた値すべてからなる集合が所望の共通集合 $A \cap B$ である．

プロトコル 3 3-card AND プロトコルをベースとした方式

Inputs：Alice：$a = (a_1, \ldots, a_n)$, Bob：$b = (b_1, \ldots, b_n)$ とする．

Setup：Alice は $\boxed{\heartsuit}$ と $\boxed{\clubsuit}$ を各 n 枚ずつ, Bob は $\boxed{\clubsuit}$ を n 枚保有する．

(1) Alice は 1 枚表現コミットメントで表現した a を Bob に手渡す．

(2) $i = 1, \ldots, n$ に関して，Bob は秘匿置換で以下の操作を行う．

- $b_i = 0$ ならば，a_i のカードを裏にした $\boxed{\clubsuit}$ で置き換える．
- $b_i = 1$ ならば何もしない．

(3) 置換後の n 枚のカード組をすべて表にする．このとき，$i = 1, \ldots, n$ に関する $a_i \wedge b_i$ の結果を得ることができ，$a_i \wedge b_i = 1$ となる u_i すべてからなる集合が所望の共通集合 $A \cap B$ である．

したことにより，$\boxed{\heartsuit}$ の枚数から入力集合のサイズ $|A|$ および $|B|$ の情報が漏洩するが，プロトコル 2 では入力集合のサイズを公開情報として仮定していることに注意する．手順 (3) で表に開いた (a'_i, b'_i) に関し，現れるパターンは $\boxed{\clubsuit}\boxed{\clubsuit}$, $\boxed{\clubsuit}\boxed{\heartsuit}$, $\boxed{\heartsuit}\boxed{\clubsuit}$, $\boxed{\heartsuit}\boxed{\heartsuit}$ の 4 通りがあり得る．$\boxed{\heartsuit}\boxed{\heartsuit}$ の個数は出力集合のサイズ $|A \cap B|$ であるため，このパターンから出力以上の情報は得られない．また，その他の出現パターン $\boxed{\clubsuit}\boxed{\clubsuit}$, $\boxed{\clubsuit}\boxed{\heartsuit}$, $\boxed{\heartsuit}\boxed{\clubsuit}$ が，出力および $|A|$, $|B|$ 以上の情報を漏らさないことを示すため，各パターンの数が $|A|$, $|B|$, $|A \cap B|$ から導出できる情報であることを示す．. $\boxed{\clubsuit}\boxed{\clubsuit}$ の数は，$i = 1, \ldots, n$ に関し $u_i \notin A$ かつ $u_i \notin B$ となるような u_i の個数である．従って，$\boxed{\clubsuit}\boxed{\clubsuit}$ の数は $|U| - (|A| + |B| - |A \cap B|)$ である．同様に，$\boxed{\clubsuit}\boxed{\heartsuit}$ および $\boxed{\heartsuit}\boxed{\clubsuit}$ の数はそれぞれ，$|B| - |A \cap B|$ および $|A| - |A \cap B|$ である．従って，手順 (4) では $A \cap B$ の要素以外のカードを表にしないことから，Alice, Bob は自身の入出力および $|A|$, $|B|$ 以上の情報を得ることはできない．

4. 秘匿置換ベースの PSI プロトコル

4.1 AND プロトコルを用いた方式

本節では，3.1 章で説明した AND プロトコルを用いて PSI プロトコルを構成する手法を，秘匿置換ベースモデルで実現する．プロトコル 3 にその手順を示す．

プロトコル 4 逐次比較に基づく方式

Inputs: Alice: $a = (a_1, \ldots, a_n)$, Bob: $b = (b_1, \ldots, b_n)$

Setup: Alice は，$\boxed{\clubsuit}$ と $\boxed{\heartsuit}$ を各 $n + 1$ 枚ずつ所持する．

1) Alice は入力値 a をコミットメント（計 $2n$ 枚）で表現する (このカード組を比較用カード組と呼ぶ)．残り 2 枚は $\boxed{\clubsuit}\boxed{\heartsuit}$ 並べて裏にする（このカード組を余分カードと呼ぶ)．

2) $i = 1, \ldots, n$ に関して，以下の操作をを行う．

2-i) Alice は比較用カード組 ($2n - 2i + 2$ 枚) および余分カード組 (2 枚) を Bob へ手渡す．

2-ii) Bob は秘匿置換を用いて以下の操作を行う．

- $b_i = 0$ ならば，比較用カード組の先頭余分カードの 2 枚を i-bit 目の出力用カード組 (p_i とする) として取り出す．
- $b_i = 1$ ならば，余分カードを i-bit 目の出力用カード組として取り出し (p_i とする)，比較用カード組の先頭の 2 枚を新たな余分カードとする．

2-iii) Bob は，一様ランダムに決定した $r_{B,i}$ に従い，秘匿置換を用いて以下の操作を行う．

- $r_{B,i} = 0$ ならば，何もしない．
- $r_{B,i} = 1$ ならば，余分カード組の並びを逆にする．

2-iv) Bob は余分カード組を Alice へ手渡す．

2-v) Alice は，一様ランダムに決定した $r_{A,i}$ に従い，秘匿置換を用いて以下の操作を行う．

- $r_{A,i} = 0$ ならば，何もしない．
- $r_{A,i} = 1$ ならば，余分カード組の並びを逆にする．

2-vi) Alice は，余分カード組を表に開き，$\boxed{\clubsuit}\boxed{\heartsuit}$ の並びにして裏にする．

3) 出力用カード組 $(p_1 \ldots, p_n)$ をすべて表にし，$p_i = \boxed{\heartsuit}\boxed{\clubsuit}$ となる u_i すべてからなる集合が所望の共通集合 $A \cap B$ である．

プロトコル 3 の Correctness および Security に関する議論は，プロトコル 1 と同様であるため省略する．

4.2 逐次比較の場合の方式

本節では，プロトコル 3 をカード枚数の観点で効率化する方式を提案する．カード枚数削減のアイデアは，プロトコル 3 で各ビットの比較で不要となったカードの再利用である．プロトコル 3 の手順 (2) に着目する．Bob は b_i に従い，$\boxed{\clubsuit}$ で置換するかしないかの操作を行うが，その結果 1 枚の不要なカードが生まれる．本節で提案するプロトコルでは，この不要なカードを再利用する．プロトコル 3 では各ビットの比較を同時に行ったが，本節では不要カードを再利用するため，各ビットの比較を逐次的に行う．結果として，カード枚数は $2n + 2$ 枚にまで削減することができた．プロトコル 4 に具体的な手順を示す．

Correctness： $b_i = 0$ となるような i については，$u_i \notin A \cap B$ であるため，手順 (3) では常に $\rho_i = \boxed{\clubsuit}\boxed{\heartsuit}$ となるべきである．出力用カードを決定する手順 2-ii) において，$b_i = 0$ ならば余分カード $\boxed{\clubsuit}\boxed{\heartsuit}$ が ρ_i として設定されるため，正しい出力を得ることができる．一方，$b_i = 1$ の場合，$a_i = 0$ ならば $u_i \notin A \cap B$, $a_i = 1$ ならば $u_i \in A \cap B$ である．従って，この場合，ρ_i には a_i を表すカードを設

定するべきである．出力用カードを決定する手順 2-ii) において，$b_i = 1$ ならば a_i のコミットメントが ρ_i として設定されるため，正しい出力を得ることができる．

Security：Bob は手順 2-ii) で自身の入力値 b_i に従い，操作を行うが操作内容は秘匿置換の仮定より秘匿されているため，Alice に b_i の情報が漏れることはない．また，手順 2-vi) で余分カードを表に開くが，このカード組は，手順 2-iii) および手順 2-v) で，Alice，Bob がそれぞれプライベートに生成した乱数 $r_{B,i}, r_{A,i}$ でランダマイズされているため，情報が漏れることはない．従って，手順 2-vi) 以外でカードを表に開く手順はないため，Alice，Bob は自身の入出力以上の情報を得ることはできない．

5. まとめ

本稿では，カードベース暗号として初めて PSI に焦点をあて，カードベース暗号における 2 つの操作モデル「シャッフルベースモデル」および「秘匿置換ベースモデル」のそれぞれで，実現手法を提案した．

初めに，それぞれのモデルにおいて，既存のカードベース AND プロトコルを用いることで PSI を実現できることを示した．その後，それぞれのモデルにおいて，この AND プロトコルをベースとした方式を効率化する方式を提案した．

シャッフルベースモデルでは，入力集合のサイズ情報の漏洩を許容することで，カード枚数を $5n$ から $3n$ へ削減し，シャッフル回数を n 回からたった 1 回にまで削減することができた (n は全体集合のサイズ)．秘匿置換ベースモデルでは，逐次比較のアイデアを導入することで，AND プロトコルをベースとした方式では破棄されていたカードを再利用することを可能とし，カード枚数 $3n$ 枚から $2n + 2$ 枚まで削減することに成功した．

謝辞　本研究は JSPS 科研費 JP21H03441, JP21H03395, JP21K17702, JP20J21248, JP18H05289, JP18H03238, JP18K19780, 17H01752 の助成，および文部科学省の卓越研究員事業の支援を受けたものです．

参考文献

[1] Bert den Boer. More efficient match-making and satisfiability: *The Five Card Trick*. In Jean-Jacques Quisquater and Joos Vandewalle, editors, *Advances in Cryptology - EUROCRYPT '89, April 10-13, 1989, Proceedings*, volume 434 of *Lecture Notes in Computer Science*, pages 208–217. Springer, 1989.

[2] Michael J. Freedman, Kobbi Nissim, and Benny Pinkas. Efficient private matching and set intersection. In *Advances in Cryptology - EUROCRYPT 2004*, pages 1–19. Springer, 2004.

[3] Rie Ishikawa, Eikoh Chida, and Takaaki Mizuki. Efficient card-based protocols for generating a hidden random permutation without fixed points. In Cristian S. Calude and Michael J. Dinneen, editors, *Unconventional Computation and Natural Computation - 14th International Conference, UCNC 2015, Auckland, New Zealand, August 30 - September 3, 2015, Proceedings*, volume 9252 of *Lecture Notes in Computer Science*, pages 215–226. Springer, 2015.

[4] Antonio Marcedone, Zikai Wen, and Elaine Shi. Secure dating with four or fewer cards. Cryptology ePrint Archive, Report 2015/1031, 2015.

[5] Takaaki Mizuki and Hideaki Sone. Six-card secure AND and four-card secure XOR. In Xiaotie Deng, John E. Hopcroft, and Jinyun Xue, editors, *Frontiers in Algorithmics, Third International Workshop, FAW 2009, Hefei, China, June 20-23, 2009. Proceedings*, volume 5598 of *Lecture Notes in Computer Science*, pages 358–369. Springer, 2009.

[6] Takeshi Nakai, Yuuki Tokushige, Yuto Misawa, Mitsugu Iwamoto, and Kazuo Ohta. Efficient card-based cryptographic protocols for millionaires' problem utilizing private permutations. In Sara Foresti and Giuseppe Persiano, editors, *Cryptology and Network Security - 15th International Conference, CANS 2016, Milan, Italy, November 14-16, 2016, Proceedings*, volume 10052 of *Lecture Notes in Computer Science*, pages 500–517, 2016.

[7] Takaaki Mizuki, Michihito Kumamoto, and Hideaki Sone. The five-card trick can be done with four cards. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, volume 7658 of *Lecture Notes in Computer Science*, pages 598–606. Springer, 2012.