Vol.2021-CSEC-93 No.5
Vol.2021-IOT-53 No.5
2021/5/13

部分開示を用いるトランプカードプロトコルとその発展

小山 寛人[1]　宮原 大輝[1,2,a)]　水木 敬明[1,2]

概要：カードベース暗号プロトコルは秘密計算を手軽に実現できる現状唯一の手法であり、その実用性の向上は秘密計算技術の普及にも繋がる。カードベース暗号分野では大抵、黒と赤の 2 色デッキを用いるプロトコルが提案されているが、市販のトランプカードデッキを用いるプロトコルもいくつか提案されている。しかし、トランプカードプロトコルの実行に必要なシャッフル回数は、2 色カードプロトコルに比べて大きいという欠点が存在していた。著者らは電子情報通信学会総合大会 2021 にて、トランプカードの絵柄のみを開示する「部分開示」操作を新たに導入することで、最小カード枚数かつ最小シャッフル回数なトランプカードプロトコルを構成した。本稿では、その提案プロトコルを理論の枠組みで形式的に記述するべく、部分開示を用いるカードベースプロトコルの計算モデルを構築する。部分開示の実現性と安全性については、その安全な実装例を図示する。部分開示を用いるプロトコルの発展として、提案プロトコルと 2 色カードプロトコルの関係を考察し、カードをめくる操作の代わりに部分開示を用いると、任意の 2 色カードプロトコルをトランプカードで実装できることを示す。さらに、2 色デッキの上下非対称性を利用すると、提案プロトコルを 2 色デッキで実装できることも示す。

Card-based Protocols Using Standard Deck of Cards Based on Half-open and Their Extensions

1. はじめに

物理的なカード組とシャッフル操作を用いて秘密計算などの暗号機能を実現する手法は**カードベース暗号**と呼ばれる。秘密計算は機密データを暗号化したまま解析することを可能にする暗号技術であり、現在実用化が進みつつある。秘密計算の実現には通常、準同型暗号や秘密分散などの暗号技術が必要であり、その仕組みを理解するためには代数学に関する知識が必要である。カードベース暗号はカード組の物理的な性質（裏になったカードの表面は見えない等）に基づいて秘密計算を実現するため、コンピュータを用いずに人間の手だけで実行できる特徴がある。カードベース暗号は秘密計算を手軽に実現できる現状唯一の手法であり、秘密計算技術の浸透に繋げるために必要不可欠な分野である [3, 7]。暗号に関する日本最大のシンポジウムである SCIS2021[*1] では、物理暗号技術に関するセッションが 2 つ設けられるほど活発に研究されている。

1.1 カードベース暗号

カードベース暗号では主に、裏面が ? のように区別がつかず、表面は黒 ♣ と赤 ♥ のいずれかである 2 色のカード組が用いられてきた。これまでに多くの研究成果（例えば [1, 2, 4, 6, 11, 13, 14, 16, 18, 20]）が得られているが、その中でも Five-card Trick と呼ばれるプロトコル [2] が最も有名かつ手軽であり、暗号に関する教科書 [17] でも紹介されている。しかし、2 色のカード組は一般的に市販されていないため、プロトコルを実行するには自作のカード組を（もしくは市販のトランプカード組を数セット）準備する必要がある。

以上を踏まえると、方式が手軽かつ市販のトランプカード組 1 セット（52 枚）だけで実行できるプロトコルの開発が望まれる。トランプカード組を用いるプロトコル [5, 8, 10, 15, 19] はいくつか存在するが、2 色カード組を用いるプロトコルと比べると、計算に必要なシャッフル回

[1] 東北大学
Tohoku University, Sendai, Miyagi 980–8579, Japan
[2] 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology (AIST)
[a)] daiki.miyahara.q4@dc.tohoku.ac.jp

[*1] https://www.iwsec.org/scis/2021/session.html

表 1: 既存のトランプカード AND プロトコルと提案プロトコルの性能

	カード枚数	シャッフル回数	部分開示回数
Niemi & Renvall [15]	5	9.5 (exp.)	0
Mizuki [10]	8	4	0
Koch & Schrempp & Kirsten [5]	4	6 (exp.)	0
Ours	4	1	1

数（すなわち結果を得るまでに必要な時間）が大きくなる欠点があった。

著者らは電子情報通信学会総合大会 2021 にて、トランプカードのスートのみを開示する**部分開示**操作を導入し、2 入力の論理積を計算するプロトコルを速報的に発表した [22]。表 1 に示すように、この提案プロトコルに必要なシャッフル回数は 1 回であり、部分開示操作も 1 回必要ではあるが、実行時間の観点から最も効率的[*2]であると言える。以上より、部分開示操作はトランプカードプロトコルの効率性を向上させ、論理関数を含む様々な計算に展開できる可能性を有する。[*3]

1.2 貢献

本稿ではまず初めに、速報的に発表した提案プロトコル [22] の計算原理を明確にするために、その構成を見直し、より分かりやすく提案プロトコルを紹介する。加えて、部分開示操作の実装例を示し、提案プロトコルが安全に実装可能であることを示す。また、提案プロトコルを基に、部分開示操作を用いるカードベース暗号プロトコルを理論の枠組みで形式的に記述できる計算モデルを構築する。

部分開示操作を用いるプロトコルの発展として、提案プロトコルと 2 色カードプロトコルの関係を考察し、部分開示操作の優位性を示す。カードをめくる操作の代わりに部分開示操作を用いると、トランプカードの表面には常に（数字とスートの組ではなく）スートのみが現れるため、トランプカード組を 4 色カード組（♣, ♠, ♥, ♦）とみなすことができる。したがって、26 枚以下の枚数を用いる 2 色カードプロトコルをトランプカード組 1 セットで実装できる。さらに、2 色デッキの上下非対称性を利用すると、提案プロトコルを 2 色デッキで実装できることも示す。

2. 提案プロトコル

本節では、総合大会 2021 で提案した部分開示操作を用いる 2 入力の AND プロトコル [22] を紹介する。この提案プロトコルの原理を説明することも本稿の貢献に含まれる。まず初めに準備として、用いるトランプカード組と部分開示操作について述べる。

2.1 準備

既存のトランプカードプロトコル [5, 8, 10, 15, 19] では、トランプカード組を 1 から 52 までの数字が書かれた 52 枚の数字カード組とみなして計算を行っていた。本稿ではトランプカード組を 13 種類の数字と 4 種類のスートの組み合わせから成る 52 枚のカード組とし、本節では次の記号を用いる。

$$\boxed{\mathrm{A}_{\clubsuit}}\boxed{\mathrm{A}_{\spadesuit}}\boxed{\mathrm{A}_{\heartsuit}}\boxed{\mathrm{A}_{\diamondsuit}}\boxed{2_{\clubsuit}}\boxed{2_{\spadesuit}}\boxed{2_{\heartsuit}}\boxed{2_{\diamondsuit}}\cdots\boxed{\mathrm{K}_{\clubsuit}}\boxed{\mathrm{K}_{\spadesuit}}\boxed{\mathrm{K}_{\heartsuit}}\boxed{\mathrm{K}_{\diamondsuit}}$$

裏面は全て同一 $\boxed{?}$ とする。

裏になったカード列 $\boxed{?}\boxed{?}\cdots\boxed{?}$ に対して、ポーカーなどのカードゲームで広く用いられている並び替え・めくる・シャッフル操作をカードベースプロトコルでも用いる。これら 3 つの操作に加え、トランプカードのスートのみを開示する部分開示操作を新たに導入する。すなわち、部分開示操作を $\boxed{?}$ に適用すると、そのカードのスートのみが明らかになり、数字は秘匿されたままである。

$$\boxed{?} \xrightarrow{\text{部分開示}} \boxed{?_{\heartsuit}} \text{ or } \boxed{?_{\diamondsuit}} \text{ or } \boxed{?_{\clubsuit}} \text{ or } \boxed{?_{\spadesuit}}$$

部分開示操作の安全な実装例を 3 節で示す。

2.2 構成

提案プロトコルは $\boxed{\mathrm{A}_{\heartsuit}}\boxed{2_{\clubsuit}}\boxed{3_{\clubsuit}}\boxed{4_{\heartsuit}}$ の 4 枚を用いて、2 つのビット $a, b \in \{0, 1\}$ から論理積 $a \wedge b$ のみを出力する。

$$\boxed{\mathrm{A}_{\heartsuit}}\boxed{2_{\clubsuit}}\boxed{3_{\clubsuit}}\boxed{4_{\heartsuit}} \rightarrow \cdots \rightarrow a \wedge b$$

ビット a を Alice が持ち、ビット b を Bob が持つものとする。

(1) Alice は $\boxed{\mathrm{A}_{\heartsuit}}\boxed{2_{\clubsuit}}$ を手に持つ。$a = 0$ であれば $\boxed{\mathrm{A}_{\heartsuit}}\boxed{2_{\clubsuit}}$ の順番で、$a = 1$ であれば $\boxed{2_{\clubsuit}}\boxed{\mathrm{A}_{\heartsuit}}$ の順番で 2 枚のカード列を裏にして置く。

$$\boxed{?}\boxed{?} \begin{cases} \boxed{\mathrm{A}_{\heartsuit}}\boxed{2_{\clubsuit}} & \text{if } a = 0 \\ \boxed{2_{\clubsuit}}\boxed{\mathrm{A}_{\heartsuit}} & \text{if } a = 1 \end{cases}$$

(2) Bob は $\boxed{3_{\clubsuit}}\boxed{4_{\heartsuit}}$ を手に持つ。Alice が置いた 2 枚のカー

[*2] シャッフル操作 1 回の実行に必要な時間を約 30 秒と見積り、カードベース暗号の実行時間を測定した研究 [9] がある。部分開示操作はシャッフル操作よりも簡便に実行できるため、部分開示操作の実行に必要な時間を加味しても提案プロトコルは効率的である。

[*3] 文献 [23] では、**一部開示テクニック**という考え方が導入されている。これは、例えば本来 5 枚のカードを全てめくることが想定されているところ、2 枚しかめくらない、というような手法である。

Vol.2021-CSEC-93 No.5
Vol.2021-IOT-53 No.5
2021/5/13

表 2: 提案プロトコルの原理

(a,b)	カード列	$A_\heartsuit$ の右	$4_\heartsuit$ の左	$2_\clubsuit$ の左	$3_\clubsuit$ の右
(0,0)	$A_\heartsuit$ $2_\clubsuit$ $3_\clubsuit$ $4_\heartsuit$	$2_\clubsuit$	$3_\clubsuit$	$A_\heartsuit$	$4_\heartsuit$
(0,1)	$A_\heartsuit$ $2_\clubsuit$ $4_\heartsuit$ $3_\clubsuit$	$2_\clubsuit$	$2_\clubsuit$	$A_\heartsuit$	$A_\heartsuit$
(1,0)	$2_\clubsuit$ $A_\heartsuit$ $3_\clubsuit$ $4_\heartsuit$	$3_\clubsuit$	$3_\clubsuit$	$4_\heartsuit$	$4_\heartsuit$
(1,1)	$2_\clubsuit$ $A_\heartsuit$ $4_\heartsuit$ $3_\clubsuit$	$4_\heartsuit$	$A_\heartsuit$	$3_\clubsuit$	$2_\clubsuit$

ド列の右横に、$b=0$ であれば $3_\clubsuit$ $4_\heartsuit$ の順番で、$b=1$ であれば $4_\heartsuit$ $3_\clubsuit$ の順番で 2 枚のカード列を裏にして置く。

$$[?][?][?][?] \begin{cases} A_\heartsuit\ 2_\clubsuit\ 3_\clubsuit\ 4_\heartsuit & \text{if } (a,b)=(0,0) \\ A_\heartsuit\ 2_\clubsuit\ 4_\heartsuit\ 3_\clubsuit & \text{if } (a,b)=(0,1) \\ 2_\clubsuit\ A_\heartsuit\ 3_\clubsuit\ 4_\heartsuit & \text{if } (a,b)=(1,0) \\ 2_\clubsuit\ A_\heartsuit\ 4_\heartsuit\ 3_\clubsuit & \text{if } (a,b)=(1,1) \end{cases}$$

このとき、$a=b=1$（すなわち $a\wedge b=1$）の場合のみ中央 2 枚のカードが $A_\heartsuit$ $4_\heartsuit$ となり、スート ♥♥ が連続していることに注意しよう。すなわち、$a=0$ の場合、Alice が置いた $A_\heartsuit$ の右にあるカードは $2_\clubsuit$ である。$a=1$ の場合、$A_\heartsuit$ の右にあるカードは、$b=0$ ならば $3_\clubsuit$ であり、$b=1$ ならば $4_\heartsuit$ である。したがって、$A_\heartsuit$ の右にあるカードのスートが $a\wedge b$ の値に対応し、そのスートだけを確認できれば $a\wedge b$ の値のみを得られる。同様に、Bob が置いた $4_\heartsuit$ の左にあるカードのスートが $a\wedge b$ の値に対応している（$a\wedge b=1$ のとき ♥、$a\wedge b=0$ のとき ♣）。

Alice が置いた $2_\clubsuit$ の（巡回的に）左にあるカードについても同じことが成り立ち、$a\wedge b=1$ の場合のみ、そのカードのスートは ♣ である。同様に、Bob が置いた $4_\heartsuit$ の（巡回的に）右にあるカードのスートは、$a\wedge b=1$ の場合のみ ♣ である。

以上の関係を表 2 にまとめる。提案プロトコルはこの関係を利用して論理積を計算する。

(3) Alice と Bob が置いた 4 枚のカード列を巡回的にシャッフルする。このシャッフル操作は**ランダムカット**と呼ばれ、$\langle\cdots\rangle$ の記号で表す。

$$\langle [?][?][?][?] \rangle \rightarrow [?][?][?][?]$$

ランダムカットは Five-card Trick [2] など多くのプロトコルに用いられているシャッフル操作であり、人間の手で安全に実装できることが示されている [21]。巡回的にカード列をシャッフルするため、表 2 に示した関係は維持されることに注意しよう。

(4) カード列の 1 枚目をめくる。

$$[?][?][?][?] \rightarrow \begin{cases} A_\heartsuit\ ?\ ?\ ? \\ 2_\clubsuit\ ?\ ?\ ? \\ 3_\clubsuit\ ?\ ?\ ? \\ 4_\heartsuit\ ?\ ?\ ? \end{cases}$$

前ステップでランダムカットを適用したため、めくる 1 枚目のカードは等確率で $A_\heartsuit$ $2_\clubsuit$ $3_\clubsuit$ $4_\heartsuit$ の内の 1 枚であり、入力に関する情報は漏れない。

(5) 次のように部分開示操作を行うことで $a\wedge b$ の値を得る（$A_\heartsuit$ か $3_\clubsuit$ のときは右のカード、それ以外のときは左のカードを部分開示）。

$$\begin{cases} A_\heartsuit\ ?_\clubsuit\ ?\ ? \\ 2_\clubsuit\ ?\ ?\ ?_\heartsuit \\ 3_\clubsuit\ ?_\heartsuit\ ?\ ? \\ 4_\heartsuit\ ?\ ?\ ?_\clubsuit \\ \rightarrow a\wedge b=0 \end{cases} \text{ or } \begin{cases} A_\heartsuit\ ?_\heartsuit\ ?\ ? \\ 2_\clubsuit\ ?\ ?\ ?_\clubsuit \\ 3_\clubsuit\ ?_\clubsuit\ ?\ ? \\ 4_\heartsuit\ ?\ ?\ ?_\heartsuit \\ \rightarrow a\wedge b=1 \end{cases}$$

すなわち、前ステップでめくったカードと同じスートが部分開示されると $a\wedge b=1$ である。

提案プロトコルに必要なシャッフル回数はランダムカット 1 回であり、必要なカード枚数は 4 枚である。既存プロトコルとの比較を含む提案プロトコルの考察を 5 節で行う。

3. 安全な実装例

本節では、著者らが提案する**部分開示**操作の実装方法を複数示す。提案方法に共通するのは、トランプカードの数字に対応する部分を追加道具などで隠しながらカードをめくることである。

スートのみが現れているトランプカードを図 1 に示す。典型的なトランプカードでは左上と右下に数字とスートが描かれていることに注目すると、次のように図 1 の状況を作り出せる。初めに、部分開示を行う裏になったカードに対し、他の（表の）カードを下に挿入し、2 枚のカードを重ねたままひっくり返す。そして数字を指で隠したままカードをずらすと、スートのみを開示できる。しかし、著者らがこの実装方法を検討してみると、実行するのに練習が必要であると判明したため、より簡便な方法を以下で考える。

図 2 に示すように、典型的なトランカード組において、4 から 10 までの数字が書かれたカードには、スートの種類

図 1: 指による部分開示操作

に依らず同じ位置（すなわち左下）にスートが描かれていることを発見した。したがって、図 3 に示すように、左下角が折りたたまれた紙をカバーにしてトランプカードをめくれば、カードの数字を隠したままスートだけを簡単に開示できる。図 3 では A7 サイズのメモ帳の紙を用いているが、トランプカードの左下のスートのみが見えれば任意のサイズの紙を用いて良い。

このカバーを用いた部分開示操作の詳細な実装方法を図 4 に示す。すなわち、部分開示を行うカードの下にカバーを挿入し（図 4a ）、2 枚を一緒に持ち上げ（図 4b ）、それらをひっくり返せば良い（図 4c ）。以上のように、部分開示操作は簡便に実装でき、カバーの作成も容易である。

4. 計算モデル

本節では、**部分開示**操作に基づくトランプカードプロトコルの計算モデルを構築する。既存の研究 [6, 12] では、カードデッキをシンボル集合上の多重集合として表し、典型的に $[1, 2, \ldots, d]$（$d \in \mathbb{N}$）としていた。一方で提案プロトコルは既存研究の枠組みには収まらず、2.1 節で述べたように、トランプカードの表面には数字に加えスートも描かれている事実を利用している。このことから、以降では**トランプカードデッキ**を組 $\mathcal{D} = (N_d, \mathrm{ss})$ と表記し、ここで $N_d = \{1, 2, \ldots, d\}$ であり、ss は N_d から $\{\clubsuit, \spadesuit, \heartsuit, \diamondsuit\}$ のようなスートシンボル集合への写像である。

4.1 表記法

デッキ (N_d, ss) に含まれる 1 や 2 などの任意の要素 $c \in N_d$ を**原子カード**と呼ぶ。$c \in N_d$ に対して、**表**のカードと**裏**のカードをそれぞれ $\frac{c}{?}$ と $\frac{?}{c}$ と表す。加えて、部分開示された裏のカードを $\frac{\mathrm{ss}(c)}{c}$ と表す。これら 3 種類のカードの原子カードを $\mathrm{atom}(\frac{c}{?}) = \mathrm{atom}(\frac{?}{c}) = \mathrm{atom}(\frac{\mathrm{ss}(c)}{c}) = c$ と表し、可視シンボルをそれぞれ $\mathrm{top}(\frac{c}{?}) = c$、$\mathrm{top}(\frac{?}{c}) = ?$、$\mathrm{top}(\frac{\mathrm{ss}(c)}{c}) = \mathrm{ss}(c)$ と表す。デッキ (N_d, ss) の d 枚のカードから成る d 項の組 $\Gamma = (\alpha_1, \alpha_2, \ldots, \alpha_d)$ を $(\{\mathrm{atom}(\alpha_1), \mathrm{atom}(\alpha_2), \ldots, \mathrm{atom}(\alpha_d)\} = N_d$ が成立する場合）**カード列**と呼ぶ。

デッキ $\mathcal{D} = (N_d, \mathrm{ss})$ から成る全ての（可能な）カード列の集合を次のように定める。

$$\mathsf{Seq}^{\mathcal{D}} := \{\Gamma \mid \Gamma \text{ is a sequence of } \mathcal{D}\}$$

$\mathrm{top}(\cdot)$ をカード列にも拡張し、カード列 $\Gamma = (\alpha_1, \alpha_2, \ldots, \alpha_d)$ に対して、$\mathrm{top}(\Gamma) = (\mathrm{top}(\alpha_1), \mathrm{top}(\alpha_2), \ldots, \mathrm{top}(\alpha_d))$ とし、これを Γ の**可視列**と呼ぶ。同様に**可視列集合** $\mathsf{Vis}^{\mathcal{D}}$ を次のように定める。

$$\mathsf{Vis}^{\mathcal{D}} := \{\mathrm{top}(\Gamma) \mid \Gamma \in \mathsf{Seq}^{\mathcal{D}}\}$$

4.2 プロトコル

部分開示操作を含むプロトコルを厳密に記述する。下で見るように、プロトコルは初期カード列から開始し、内部状態と可視列に応じて現カード列に適用する動作をステップ毎に決定する。

（**有限状態制御部**とカード列が置かれる**テーブル**を有する）**プロトコル**は次の 4 項組 $\mathcal{P} = (\mathcal{D}, U, Q, A)$ によって定まる。

- $\mathcal{D} = (N_d, \mathrm{ss})$ はデッキである。
- $U \subseteq \mathsf{Seq}^{\mathcal{D}}$ は**入力集合**である。
- Q は初期状態 $q_0 \in Q$ と終了状態 $q_{\mathrm{f}} \in Q$ を持つ**状態集合**である。
- $A : (Q \backslash \{q_{\mathrm{f}}\}) \times \mathsf{Vis}^{\mathcal{D}} \to Q \times \mathsf{Action}$ は**動作関数**であり、Action は次の動作から成る集合である。
  - $T \subseteq \{1, 2, \ldots, d\}$ に対して、(turn, T)。
  - d 次の対称群 S_d に含まれる置換 $\pi \in S_d$ に対して、(perm, π)。
  - 置換集合 $\Pi \subseteq S_d$ と Π 上の確率分布 $\mathcal{F}$ に対して、$(\mathrm{shuf}, \Pi, \mathcal{F})$。もし $\mathcal{F}$ が一様である場合、$\mathcal{F}$ を省略して (shuf, Π) と書く。
  - $T \subseteq \{1, 2, \ldots, d\}$ に対して、(hopen, T)。
  - $T \subseteq \{1, 2, \ldots, d\}$ に対して、(hclose, T)。

カード列 $\Gamma = (\alpha_1, \alpha_2, \ldots, \alpha_d)$ に対して、Action に含まれるそれぞれの動作は、次のように Γ を Γ' に遷移させる。

- (turn, T): 全ての $i \in \{1, \ldots, d\}$ に対して $\Gamma' = (\beta_1, \beta_2, \ldots, \beta_d)$ かつ

$$\beta_i = \begin{cases} \mathrm{swap}(\alpha_i) & \text{if } i \in T \\ \alpha_i & \text{otherwise} \end{cases}$$

  ここで原子カード c に対して、$\mathrm{swap}(\frac{c}{?}) = \frac{?}{c}$ であり、$\mathrm{swap}(\frac{?}{c}) = \frac{c}{?}$ である。
- (perm, π): $\Gamma' = (\alpha_{\pi^{-1}(1)}, \alpha_{\pi^{-1}(2)}, \ldots, \alpha_{\pi^{-1}(d)})$ である。
- $(\mathrm{shuf}, \Pi, \mathcal{F})$: 確率分布 $\mathcal{F}$ に従って Π から選ばれた置換 π に対して、Γ' は (perm, π) を Γ に適用したカード列となる。

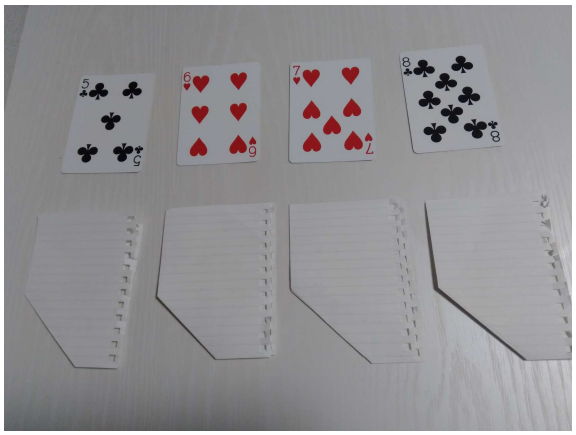


図 2: 典型的なトランプカード

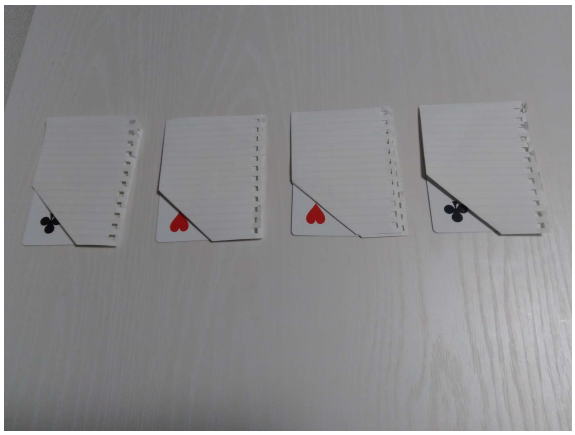


図 3: カバーを使用

(a) カードの下にカバーを挿入

(b) 2 つのカードを一緒に持つ

(c) めくる

図 4: カバーを用いた部分開示操作の実装

- (hopen, T): 全ての $i \in \{1,\dots,d\}$ に対して、$\Gamma' = (\beta_1, \beta_2, \dots, \beta_d)$ かつ

$$\beta_i = \begin{cases} \frac{\mathsf{ss}(\mathsf{atom}(\alpha_i))}{\mathsf{atom}(\alpha_i)} & \text{if } i \in T, \\ \alpha_i & \text{otherwise} \end{cases}$$

  ここで全ての $j \in T$ に対して、α_j は裏のカード $\alpha_j = \frac{?}{\mathsf{atom}(\alpha_j)}$ である。
- (hclose, T): 全ての $i \in \{1,\dots,d\}$ に対して、$\Gamma' = (\beta_1, \beta_2, \dots, \beta_d)$ かつ

$$\beta_i = \begin{cases} \frac{?}{\mathsf{atom}(\alpha_i)} & \text{if } i \in T \\ \alpha_i & \text{otherwise} \end{cases}$$

  ここで全ての $j \in T$ に対して、α_j は既に部分開示された裏のカード $\alpha_j = \frac{\mathsf{ss}(\mathsf{atom}(\alpha_j))}{\mathsf{atom}(\alpha_j)}$ である。

(hopen, T) と (hclose, T) が新たに導入された動作であり、それぞれ部分開示とそれを戻す動作である。

4.3 提案プロトコルの疑似コード

4.2 節で定式化した計算モデルを基に、2.2 節で紹介した提案プロトコル [22] の疑似コードをアルゴリズム 1 に示す。ここで写像 ss と入力集合 U は次の通りである。

$\mathsf{ss} : A \mapsto ♥,\ 2 \mapsto ♣,\ 3 \mapsto ♣,\ 4 \mapsto ♥$

$$U = \left\{ \left(\frac{?}{A}, \frac{?}{2}, \frac{?}{3}, \frac{?}{4}\right), \left(\frac{?}{A}, \frac{?}{2}, \frac{?}{4}, \frac{?}{3}\right), \left(\frac{?}{2}, \frac{?}{A}, \frac{?}{3}, \frac{?}{4}\right), \left(\frac{?}{2}, \frac{?}{A}, \frac{?}{4}, \frac{?}{3}\right) \right\}$$

疑似コードは約 10 行で記述できるほどシンプルである。

5. 考察

本節では、部分開示操作を用いるプロトコルの発展として、提案プロトコルと 2 色カードプロトコルの関係を考察し、部分開示操作の優位性を示す。初めに、提案プロトコルと同じ 4 枚のカードを用いて AND 計算を行う 2 色カードプロトコル [11] を取り上げ、性能を比較する。

5.1 性能比較

2012 年に Mizuki らによって提案された 2 色 AND プロトコル [11] は、4 枚のカード $\boxed{♣}\boxed{♣}\boxed{♥}\boxed{♥}$ を用いて 2 入力の AND 計算を行うプロトコルである。

$$\boxed{♣}\boxed{♣}\boxed{♥}\boxed{♥} \to \cdots \to a \wedge b$$

Mizuki らのプロトコルは、Five-card Trick [2] の実行に必要なカード枚数を 1 枚減らし、1 つのビットを 2 枚のカードで符号化するルールの下では最小の 4 枚で実行できることが特徴である。しかし、実行に必要なシャッフル回数は

Algorithm 1 提案 2 入力 AND プロトコル $((\{A,2,3,4\},\mathrm{ss}),U,Q,A)$ [22]

(1) $(\mathsf{shuf},\{(1\,2\,3\,4)^i \mid i\in\{0,1,2,3\}\})$
(2) $(\mathsf{turn},\{1\})$
(3) **if** visible seq. $=(A,?,?,?)$ or $(3,?,?,?)$ **then**
(4) $\quad(\mathsf{hopen},\{2\})$
(5) $\quad$**if** visible seq. $=(A,\heartsuit,?,?)$ or $(3,\clubsuit,?,?)$ **then** $a\wedge b=1$
(6) $\quad$**else if** visible seq. $=(A,\clubsuit,?,?)$ or $(3,\heartsuit,?,?)$ **then** $a\wedge b=0$
(7) **else if** visible seq. $=(2,?,?,?)$ or $(4,?,?,?)$ **then**
(8) $\quad(\mathsf{hopen},\{4\})$
(9) $\quad$**if** visible seq. $=(2,?,?,\clubsuit)$ or $(4,?,?,\heartsuit)$ **then** $a\wedge b=1$
(10) $\quad$**else if** visible seq. $=(2,?,?,\heartsuit)$ or $(4,?,?,\clubsuit)$ **then** $a\wedge b=0$

2 回[*4]であり、その計算原理は Five-card Trick の原理より複雑である欠点が存在する。

1.1 節で述べたように、カードベース暗号の研究ではこれまでに、トランプカードプロトコル [5,8,10,15,19] の効率性は 2 色カードプロトコルよりも劣る結果が得られていた。しかし、2.2 節で示した提案プロトコルに必要なシャッフル回数は 1 回であり、部分開示操作が 1 回必要なことを除くと、Mizuki らの 2 色カードプロトコル [11] よりも効率的である。すなわち、著者らが導入した部分開示操作に基づく提案プロトコルは、表 1 に示したように既存のトランプカードプロトコル [5,10,15] よりも効率的であるだけでなく、トランプカードプロトコルが 2 色カードプロトコルよりも効率的に実行できることを初めて示している。これは部分開示操作が有用であることを示し、AND 計算以外の様々な計算でも同様の結果が得られる可能性がある。

2.2 節で示した提案プロトコルと Mizuki らのプロトコル [11] に共通する計算原理を考察し、Five-card Trick [2] のように分かりやすいプロトコルの表現方法を見つけることが今後の課題である。

5.2 トランプカード組による 2 色カードプロトコルの実装

1.2 節で述べたように、トランプカードをめくる操作の代わりに部分開示操作を用いると、トランプカードの表面には常に（数字とスートの組ではなく）スートのみが現れるため、1 枚のトランプカードをスートのみが書かれた 1 枚のカードとみなすことができる。例えば、裏になった黒 $\boxed{\clubsuit}$ と $\boxed{\mathrm{A}_{\clubsuit}}$ は次のように同一視できる。

$$\boxed{?}\xrightarrow{\text{めくる}}\boxed{\clubsuit}\;=\;\boxed{?}\xrightarrow{\text{部分開示}}\boxed{?_{\clubsuit}}$$

すなわち、トランプカード組を 4 色カード組（$\clubsuit,\spadesuit,\heartsuit,\diamondsuit$）とみなせるため、26 枚以下の枚数を用いる 2 色カードプロトコルをトランプカード組 1 セットで実装できる。例えば、Five-card Trick [2] を実装するには、めくる操作の代わりに部分開示操作を 3 回だけ用いれば良い。まとめると、1.1 節で述べた 2 色カードプロトコルに用いるカード組は自作する必要がある欠点を、部分開示操作は解決している。

[*4] ランダムカット 1 回と、ランダム二等分割カットと呼ばれるシャッフル操作 1 回を用いる。

より顕著に解決できる例は、2020 年に発表された大小比較（金持ち比べ）に関する研究 [8] である。トランプカード組を用いる比較プロトコルに必要なカード枚数とシャッフル回数は、2 色カードプロトコルに比べてそれぞれ約 4/3 倍と 4 倍に増加するが、めくる操作の代わりに部分開示操作を 1 回行うだけで、同じカード枚数・シャッフル回数のまま大小比較を実現できることが分かった。詳細な考察は今後の課題である。

5.3 上下非対称性を利用した提案プロトコルの実装

5.1 節で述べたように、提案プロトコルは同じ 4 枚のカードを用いる Mizuki らの 2 色カードプロトコル [10] よりも効率的（シャッフル回数が 1 回少ない）であるため、2 色カード組（$\boxed{\clubsuit}\boxed{\heartsuit}$）で提案プロトコルを実装できれば Mizuki らのプロトコルよりも効率的に AND 計算を実行できる。これは 2 色カード組の表面の上下非対称性を利用すれば可能であり、具体的には Bob が置く 2 枚のカードを上下逆さま（$\boxed{\spadesuit}\boxed{\heartsuit}$）にすれば良い。

カードの裏面は上下対称である必要があるため、以降では裏になったカードを $\boxed{}$ とする。この裏になったカードに対して部分開示を行うと、次のようにそのカードの色（黒 $\boxed{\bullet}$ か赤 $\boxed{\bullet}$）のみが開示され、上下逆さまかどうかは漏れないものとする。

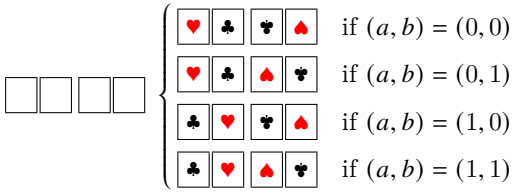
$$\boxed{}\xrightarrow{\text{部分開示}}\boxed{\bullet}\ \text{or}\ \boxed{\bullet}$$

実装方法の詳細は以下の通りである。

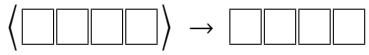
(1) Alice は $\boxed{\clubsuit}\boxed{\heartsuit}$ を手に持つ。$a=0$ であれば $\boxed{\heartsuit}\boxed{\clubsuit}$ の順番で、$a=1$ であれば $\boxed{\clubsuit}\boxed{\heartsuit}$ の順番で 2 枚のカード列を裏にして置く。

$$\boxed{}\boxed{}\begin{cases}\boxed{\heartsuit}\boxed{\clubsuit} & \text{if } a=0\\ \boxed{\clubsuit}\boxed{\heartsuit} & \text{if } a=1\end{cases}$$

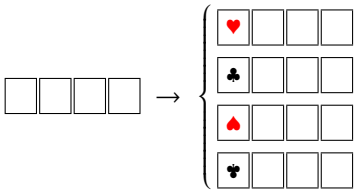
(2) Bob は $\boxed{\clubsuit}\boxed{\heartsuit}$ を手に持つ。Alice が置いた 2 枚のカード列の右横に、$b=0$ であれば $\boxed{\spadesuit}\boxed{\heartsuit}$ の順番で、$b=1$ であれば $\boxed{\heartsuit}\boxed{\spadesuit}$ の順番で 2 枚のカード列を裏にして上下逆さまに置く。



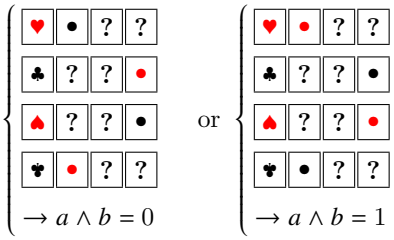
(3) 4 枚のカード列にランダムカットを適用する。



(4) カード列の 1 枚目をめくる。



(5) 次のように部分開示操作を行うことで $a \wedge b$ の値を得る（♥か♠のときは右のカード、それ以外のときは左のカードを部分開示）。



謝辞 本研究は JSPS 科研費 JP19J21153 の助成を受けたものです。

参考文献

[1] Crépeau, C. and Kilian, J.: Discreet Solitary Games, *Advances in Cryptology—CRYPTO' 93* (Stinson, D. R., ed.), Lecture Notes in Computer Science, Vol. 773, Berlin, Heidelberg, Springer, pp. 319–330 (online), available from ⟨https://doi.org/10.1007/3-540-48329-2_27⟩ (1994).

[2] Den Boer, B.: More Efficient Match-Making and Satisfiability The Five Card Trick, *Advances in Cryptology—EUROCRYPT '89* (Quisquater, J.-J. and Vandewalle, J., eds.), Lecture Notes in Computer Science, Vol. 434, Berlin, Heidelberg, Springer, pp. 208–217 (online), available from ⟨https://doi.org/10.1007/3-540-46885-4_23⟩ (1990).

[3] Hanaoka, G.: Towards User-Friendly Cryptography, *Paradigms in Cryptology–Mycrypt 2016. Malicious and Exploratory Cryptology* (Phan, R. C.-W. and Yung, M., eds.), Lecture Notes in Computer Science, Vol. 10311, Cham, Springer, pp. 481–484 (online), available from ⟨https://doi.org/10.1007/978-3-319-61273-7_24⟩ (2017).

[4] Kastner, J., Koch, A., Walzer, S., Miyahara, D., Hayashi, Y., Mizuki, T. and Sone, H.: The Minimum Number of Cards in Practical Card-Based Protocols, *Advances in Cryptology—ASIACRYPT 2017* (Takagi, T. and Peyrin, T., eds.), Lecture Notes in Computer Science, Vol. 10626, Cham, Springer, pp. 126–155 (online), available from ⟨https://doi.org/10.1007/978-3-319-70700-6_5⟩ (2017).

[5] Koch, A., Schrempp, M. and Kirsten, M.: Card-Based Cryptography Meets Formal Verification, *Advances in Cryptology–ASIACRYPT 2019* (Galbraith, S. D. and Moriai, S., eds.), Lecture Notes in Computer Science, Vol. 11921, Cham, Springer, pp. 488–517 (online), available from ⟨https://doi.org/10.1007/978-3-030-34578-5_18⟩ (2019).

[6] Koch, A., Walzer, S. and Härtel, K.: Card-Based Cryptographic Protocols Using a Minimal Number of Cards, *Advances in Cryptology—ASIACRYPT 2015* (Iwata, T. and Cheon, J. H., eds.), Lecture Notes in Computer Science, Vol. 9452, Berlin, Heidelberg, Springer, pp. 783–807 (online), available from ⟨https://doi.org/10.1007/978-3-662-48797-6_32⟩ (2015).

[7] Marcedone, A., Wen, Z. and Shi, E.: Secure Dating with Four or Fewer Cards, Cryptology ePrint Archive, Report 2015/1031 (2015).

[8] Miyahara, D., Hayashi, Y., Mizuki, T. and Sone, H.: Practical card-based implementations of Yao's millionaire protocol, *Theoretical Computer Science*, Vol. 803, pp. 207–221 (online), available from ⟨https://doi.org/10.1016/j.tcs.2019.11.005⟩ (2020).

[9] Miyahara, D., Ueda, I., Hayashi, Y., Mizuki, T. and Sone, H.: Evaluating card-based protocols in terms of execution time, *International Journal of Information Security*, pp. 1–12 (online), available from ⟨https://doi.org/10.1007/s10207-020-00525-4⟩ (2020).

[10] Mizuki, T.: Efficient and Secure Multiparty Computations Using a Standard Deck of Playing Cards, *Cryptology and Network Security* (Foresti, S. and Persiano, G., eds.), Lecture Notes in Computer Science, Vol. 10052, Cham, Springer, pp. 484–499 (online), available from ⟨https://doi.org/10.1007/978-3-319-48965-0_29⟩ (2016).

[11] Mizuki, T., Kumamoto, M. and Sone, H.: The Five-Card Trick Can Be Done with Four Cards, *Advances in Cryptology—ASIACRYPT 2012* (Wang, X. and Sako, K., eds.), Lecture Notes in Computer Science, Vol. 7658, Berlin, Heidelberg, Springer, pp. 598–606 (online), available from ⟨https://doi.org/10.1007/978-3-642-34961-4_36⟩ (2012).

[12] Mizuki, T. and Shizuya, H.: A formalization of card-based cryptographic protocols via abstract machine, *International Journal of Information Security*, Vol. 13, No. 1, pp. 15–23 (online), available from ⟨https://doi.org/10.1007/s10207-013-0219-4⟩ (2014).

[13] Mizuki, T. and Sone, H.: Six-Card Secure AND and Four-Card Secure XOR, *Frontiers in Algorithmics* (Deng, X., Hopcroft, J. E. and Xue, J., eds.), Lecture Notes in Computer Science, Vol. 5598, Berlin, Heidelberg, Springer, pp. 358–369 (online), available from ⟨https://doi.org/10.1007/978-3-642-02270-8_36⟩ (2009).

[14] Nakai, T., Misawa, Y., Tokushige, Y., Iwamoto, M. and Ohta, K.: How to Solve Millionaires' Problem with Two Kinds of Cards, *New Generation Computing*, (online), available from ⟨https://doi.org/10.1007/s00354-020-00118-8⟩ (2021, in press).

[15] Niemi, V. and Renvall, A.: Solitaire Zero-knowledge, *Fundam. Inf.*, Vol. 38, No. 1,2, pp. 181–188 (online), available from ⟨https://doi.org/10.3233/FI-1999-381214⟩ (1999).

[16] Ono, H. and Manabe, Y.: Card-Based Cryptographic Logical Computations Using Private Operations, *New Generation Computing*, (online), available from ⟨https://doi.org/10.1007/s00354-020-00113-z⟩ (2020, in press).
[17] Pass, R. and Shelat, A.: A Course in Cryptography (2010).
[18] Ruangwises, S. and Itoh, T.: Physical Zero-Knowledge Proof for Numberlink Puzzle and k Vertex-Disjoint Paths Problem, *New Generation Computing*, (online), available from ⟨https://doi.org/10.1007/s00354-020-00114-y⟩ (2020, in press).
[19] Shinagawa, K. and Mizuki, T.: Secure Computation of Any Boolean Function Based on Any Deck of Cards, *Frontiers in Algorithmics* (Chen, Y., Deng, X. and Lu, M., eds.), Lecture Notes in Computer Science, Vol. 11458, Cham, Springer, pp. 63–75 (online), available from ⟨https://doi.org/10.1007/978-3-030-18126-0_6⟩ (2019).
[20] Shinagawa, K. and Nuida, K.: A single shuffle is enough for secure card-based computation of any Boolean circuit, *Discrete Applied Mathematics*, Vol. 289, pp. 248–261 (online), available from ⟨https://doi.org/10.1016/j.dam.2020.10.013⟩ (2021).
[21] Ueda, I., Miyahara, D., Nishimura, A., Hayashi, Y., Mizuki, T. and Sone, H.: Secure implementations of a random bisection cut, *International Journal of Information Security*, Vol. 19, No. 4, pp. 445–452 (online), available from ⟨https://doi.org/10.1007/s10207-019-00463-w⟩ (2020).
[22] 小山寛人，宮原大輝，水木敬明，曽根秀昭：トランプカードの部分開示を用いた AND 秘密計算，電子情報通信学会総合大会 2021，A-7-1 (2021).
[23] 須賀祐治：手の内だけで簡単に実行可能な Six Card Trick とカード入力後の置換に関する考察，第 92 回 CSEC 研究発表会，7 (2021).