Computer Security Symposium 2020
26 - 29 October 2020

6枚のカードを用いた3入力多数決関数の秘密計算

豊田 航大[1,a)]　宮原 大輝[1,2]　水木 敬明[3]　曽根 秀昭[3]

概要：物理的なカード組を用いて秘密計算を行う手法をカードベース暗号という．カードベース暗号が対象としてきた有用な関数の 1 つに，多数決関数がある．Nishida らは初めてその具体的な構成を示し，シャッフル操作と 8 枚のカードを用いると 3 入力多数決関数を秘密計算できることを示した．その後 Nakai らによって，背面処理を導入すると 4 枚のカードで実現できることが示され，さらに Watanabe らによって，3 枚のみを用いる構成が提案されている．本論文では，背面処理を用いない設定でのプロトコルの改良を考え，6 枚のカードで 3 入力多数決関数プロトコルを構成できることを示す．提案するのは非コミット型とコミット型の両方である．非コミット型は，最も有名な five-card trick の原理を利用し，2 回のシャッフル操作で必ず終了する実用的なプロトコルである．コミット型では，Abe らが提案した 5 枚 AND プロトコルのアイデアを利用し，非コミット型からコミット型への変形を行っている．提案プロトコルは，2 枚のカードで 1 ビットを符号化する設定の下では最小のカード枚数である．実用的なシャッフル操作のみで 3 入力多数決関数プロトコルを最小枚数で構成できたことは興味深い結果である．

キーワード：秘密計算，物理的暗号技術，カードベース暗号，多数決関数

Secure Computation of Three-Input Majority Function Using Six Cards

Kodai Toyoda[1,a)]　Daiki Miyahara[1,2]　Takaaki Mizuki[3]　Hideaki Sone[3]

Abstract: A card-based protocol for securely computing the three-input majority function was first constructed by Nishida et al. using practical shuffle actions with eight cards. Nakai et al. then showed that it is possible to realize the same task with four cards by using private permutations and Watanabe et al. reduced the number of cards to three. In this paper, we seek better protocols without relying on private permutations; we propose two protocols for the three-input majority function with six cards. One is a non-committed-format protocol based on the idea behind the famous five-card trick. The other is a committed-format protocol, which is obtained by transforming our first protocol into a committed-format one thanks to the idea behind the five-card AND protocol proposed by Abe et al. Both of our protocols use the minimum number of cards under the setting where one bit is encoded with two cards.

Keywords: Secure Computation, Physical Cryptography, Card-based Cryptography, Majority Protocol

1. はじめに

トランプのようなカード組を用いて秘密計算を行う手法をカードベース暗号という．カードベース暗号では，表面が $\boxed{\clubsuit}$ または $\boxed{\heartsuit}$ であり，裏面が $\boxed{?}$ で区別のつかない 2 種類のカードを用いる．このカードを用いて次のようにブール値を表す．

$$\boxed{\clubsuit}\boxed{\heartsuit} = 0,\ \boxed{\heartsuit}\boxed{\clubsuit} = 1 \qquad (1)$$

ビット $x \in \{0,1\}$ がこの符号化ルールに従って符号化され，2 枚のカードが裏返しに置かれる時，この 2 枚のカードを x のコミットメントと呼び，次のように書く．

[1] 東北大学大学院情報科学研究科
Graduate School of Information Sciences, Tohoku University
[2] 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology
[3] 東北大学サイバーサイエンスセンター
Cyberscience Center, Tohoku University
[a)] kodai.toyoda.p1@dc.tohoku.ac.jp

コミットメントで入出力をするプロトコルを**コミット型プロトコル**という．

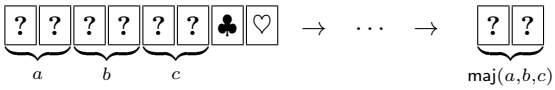
カードベース暗号が対象としてきた有用な関数の 1 つに，多数決関数がある．3 入力多数決関数 $\mathsf{maj}(a,b,c)$ は

$$\mathsf{maj}(a,b,c) = \begin{cases} 1 \text{ if } a+b+c \geq 2 \\ 0 \text{ if } a+b+c \leq 1 \end{cases}$$

と定義される．コミット型の 3 入力多数決関数プロトコルは，入力値 $a,b,c \in \{0,1\}$ の 3 つのコミットメントを受け取り，それらに関する情報を漏らすことなく $\mathsf{maj}(a,b,c)$ の値のコミットメントを出力するプロトコルである．

1.1 既存研究

3 入力多数決関数プロトコルに関する既存研究を表 1 に示す．Nishida ら [1] は初めてその具体的な構成を示し，3 つの入力コミットメント分の 6 枚に 2 枚を加え，合計 8 枚のカードを用いたコミット型の 3 入力多数決関数を秘密計算するプロトコルを示した．



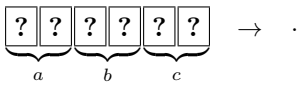
その後 Nakai ら [2] によって，背面処理を導入すると 4 枚のカードで実現できることが示され，さらに Watanabe ら [3] によって，3 枚のみを用いる構成が提案された．背面処理とは，プロトコルの実行中にあるプレイヤーが他のプレイヤーに見られないように行う操作である．Nakai らのプロトコルは，入力値に応じた背面での並び替え操作とカードをめくる操作を含んでいる．Watanabe らのプロトコルはそれに加えて，口答で出力を出し，その際に自分の入力に応じて逆の値を公開する操作，つまり背面での NOT 計算を利用する．また，最近では Yasunaga [4] により，6 枚のカードを用いたより単純な背面処理を利用したプロトコルが提案されている．このプロトコルは途中で a のコミットメントを再構成する必要があり，めくられるカードを利用して，入力コミットメントの再構成（その際に背面処理を利用する）と途中作成を行う必要がある．Yasunaga はさらに，このプロトコルにカードを 2 枚追加し，あらかじめ入力コミットメントのコピーを取ることで背面処理を用いないプロトコルも提案している．ただし，こちらも入力コミットメントの途中作成を含んでいる．これらのプロトコルは出力が式 (1) に従うコミットメントとして得られるわけではなく，非コミット型である．

表 1 既存の 3 入力多数決関数プロトコルと提案プロトコル

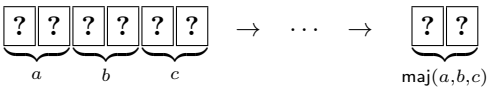
	カード枚数	背面処理	コミット型	有限
Nishida et al., 2013 [1]	8	なし	✓	✓
Nakai et al., 2017 [2]	4	2 回		✓
Watanabe et al., 2018 [3]	3	4 回		✓
Yasunaga (1), 2020 [4]	6	1 回		✓
Yasunaga (2), 2020 [4]	8	なし		✓
本論文（3 節）	**6**	なし		✓
本論文（4 節）	**6**	なし	✓	

1.2 本論文の貢献

本論文では，Nishida らの 3 入力多数決関数プロトコル [1] を改良し，6 枚のカードで構成されるプロトコルを 2 つ提案する．1 つは非コミット型プロトコルであり，3 節で述べる．



もう 1 つはコミット型プロトコルであり，4 節で述べる．



3 入力多数決関数プロトコルは，最低でも 3 つの入力コミットメント分の 6 枚のカードを必要とする．よって，提案するプロトコルは 2 つとも，符号化ルール (1) の下ではカード枚数の観点で最適である．（Yasunaga のプロトコルは c のコミットメントを途中作成していることに注意.）

提案非コミット型プロトコルは，後で紹介するランダム二等分割カットと Five-Card Trick を利用して構成され，必要なシャッフル回数は 2 回である．提案コミット型プロトコルは Abe ら [5] の AND プロトコルを利用して非コミット型をコミット型に変換することで得られる．提案コミット型プロトコルの実行時間は非有限時間ではあるものの，Koch ら [6] の AND プロトコルに必要な複雑なシャッフル操作を必要とせずに最小枚数を達成できたことは興味深い結果である．

1.3 本論文の構成

本論文の構成は次の通りである．1 節では 3 入力多数決関数の既存研究と本論文の貢献について述べた．2 節では提案プロトコルに必要な操作と Five-Card Trick について説明する．そして，3 節で提案非コミット型プロトコルについて，4 節でコミット型プロトコルについて説明する．最後に 5 節で結論を述べる．

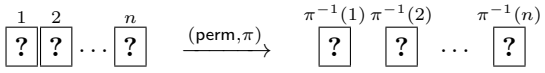
2. 準備

本節では，カードベース暗号の計算モデル [7] で用いられる操作について説明する．次にカードベース暗号における実用的なシャッフル操作である，ランダムカットとランダム二等分割カットについて説明する．さらに，最初のカードベースプロトコルである Five-Card Trick について説明する．

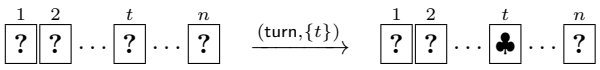
2.1 カードベース暗号で使用する操作

カードベース暗号では，カード列に対して主に 3 つの操作を行う．

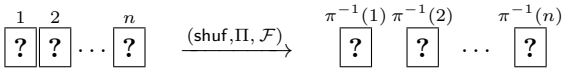
並べ替え カード列に対し置換 π を適用する．



めくる カード列の左から t 枚目のカードをめくってカードの色を確認する．



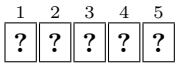
シャッフル カード列に対し置換集合 Π から確率分布 $\mathcal{F}$ に従って得られる置換 π を適用する．



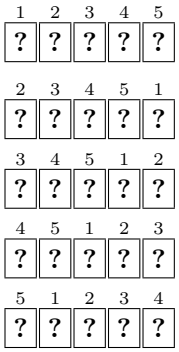
ただし，Π に含まれるどの置換が適用されたのかは誰も知り得ない．確率分布が一様である場合は，確率分布 $\mathcal{F}$ を省略する場合がある．

2.2 ランダムカット

ランダムカットとは，誰も並びが分からなくなるようにカード列を巡回的にランダムにシフトさせるシャッフル操作である．説明のために 5 枚のカード列に次のように番号を振る．



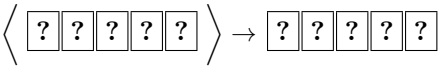
ランダムカット後のカード列は次の 5 通りのいずれかになり，生起確率はそれぞれ 1/5 である．



このランダムカットは，巡回置換 $\pi = (1\,2\,3\,4\,5)$ を用いて

$$(\mathsf{shuf}, \{\mathsf{id}, \pi, \pi^2, \pi^3, \pi^4\})$$

と書くことができる．ここで id は恒等置換である．ランダムカットは操作が簡単で，人間が安全に実行できることが実験的に確認されている [8]．ランダムカットは $\langle\cdot\rangle$ と表記する．例えば 5 枚のカードにランダムカットを適用する場合は，

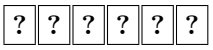


と書く．

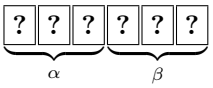
2.3 ランダム二等分割カット

ランダム二等分割カットは 2009 年に Mizuki と Sone [9] によって考案されたシャッフル法である．以下では，6 枚の場合を例として説明する．

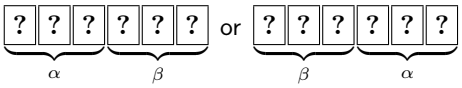
(1) 6 枚のカードが並んでいるとする．



(2) カード列を半分に分け，左半分を α，右半分を β とする．

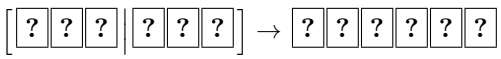


(3) α と β の位置をランダムに入れ替える．



α と β の位置はそのまま，もしくは入れ替わることとなり，その確率はそれぞれ 1/2 である．

以上がランダム二等分割カットの操作であり，$[\cdot|\cdot]$ と表記する．例えば 6 枚のカードにランダム二等分割カットを適用する場合は，



と書く．また，このランダム二等分割カットは，

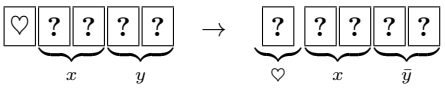
$$(\mathsf{shuf}, \{\mathsf{id}, (1\,4)(2\,5)(3\,6)\})$$

と書くことができる．ランダム二等分割カットは身近な道具を用いて安全に実装できることが知られ [8]，カードの裏面が上下非対称の場合はランダムカットを用いて実装できる [10]．

2.4 Five-Card Trick

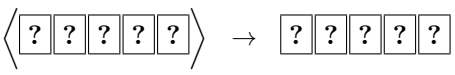
Five-Card Trick [11] は den Boer によって提案された非コミット型の AND プロトコルであり，$x, y \in \{0, 1\}$ のコミットメントと追加のカード 1 枚を入力として $x \wedge y$ を出力する．その手順は以下の通りである（文献 [11] の本質を変えずに，後の説明のために一部カードの並びを変えている）．

(1) 追加のカード $\boxed{\heartsuit}$ と 2 つの入力コミットメントを置く．y のコミットメントの左右のカードを入れ替えること（NOT 計算）により $\bar{y}$ のコミットメントを得る．追加カードは裏返す．



このとき，1 枚目，2 枚目，5 枚目の 3 枚は，$x = y = 1$ のとき，すなわち $x \wedge y = 1$ のときに限り，$\boxed{\heartsuit}\boxed{\heartsuit}\boxed{\heartsuit}$ となり，赤のカード 3 枚が巡回的に連続する．

(2) ランダムカットをカード列に適用する．



(3) 5 枚のカード全てを表にすると，3 枚の $\boxed{\heartsuit}$ が巡回的に連続して並ぶか，そうでないかのどちらかになる．前者の場合 $x \wedge y = 1$ であり，後者の場合 $x \wedge y = 0$ である．

3. 提案非コミット型プロトコル

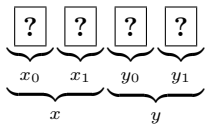
3 入力多数決関数 $\mathsf{maj}(a,b,c)$ は

$$\mathsf{maj}(a,b,c) = \begin{cases} b \wedge c \text{ if } a = 0 \\ b \vee c \text{ if } a = 1 \end{cases}$$

の関係が成り立つ．この条件を満たすようにプロトコルを構成することを目指す．$b \wedge c$ は 2.4 節の Five-Card Trick を利用して計算でき，あとは $b \vee c$ を計算する方法が必要となる．そこで本節では最初に，Five-Card Trick の操作を少し変更して，OR 計算を行うプロトコルを提案する．このとき，赤のカードのかわりに黒のカードを追加したプロトコルについても説明する．次に，提案非コミット型プロトコルのアイデアについて述べる．その後，提案非コミット型プロトコルの手順を説明する．さらに，その正当性と安全性を証明する．

3.1 Five-Card Trick のバリエーション

本節では，Five-Card Trick の考えに基づき，追加カードが $\boxed{\heartsuit}$ である場合と $\boxed{\clubsuit}$ である場合それぞれで AND を計算するプロトコルと OR を計算するプロトコルを説明する．本論文では $\boxed{\clubsuit}$ を追加カードとする場合，$\boxed{\heartsuit}$ を追加カードとした場合と逆に，$\boxed{\clubsuit}$ が 3 枚巡回的に連続している場合を出力が 0，そうでない場合を出力が 1 として計算する．$\boxed{\heartsuit}$ を追加カードとするプロトコルを赤ベースプロトコル，$\boxed{\clubsuit}$ を追加カードとするプロトコルを黒ベースプロトコルと呼ぶことにする．ここで，説明のために x と y のコミットメントを x_0, x_1, y_0, y_1 を用いて次のように書くことにする．

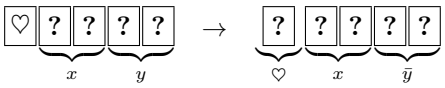


赤ベース AND

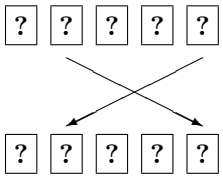
2.4 節の Five-Card Trick そのものである．

赤ベース OR

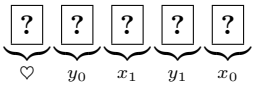
(1) 追加のカード $\boxed{\heartsuit}$ と 2 つの入力コミットメントを置き，y のコミットメントの左右のカードを入れ替えること（NOT 計算）により $\bar{y}$ のコミットメントを得る．追加カードは裏返す．



(2) カードを次のように並べかえる．



するとカード列は次のようになる．



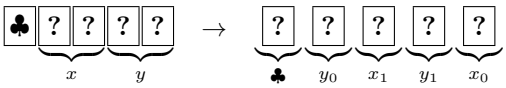
このとき，$x \vee y = 1$ のときに限り，赤のカード 3 枚が巡回的に連続することが確認できる．

(3) ランダムカットをカード列に適用する．

(4) 5 枚のカード全てを表にすると，3 枚の $\boxed{\heartsuit}$ が巡回的に連続して並ぶか，そうでないかのどちらかになる．前者の場合 $x \vee y = 1$ であり，後者の場合 $x \vee y = 0$ である．

黒ベース AND

(1) 追加カードを $\boxed{\clubsuit}$ として赤ベース OR のステップ 1 と 2 を実行する．するとカード列は次のようになる．



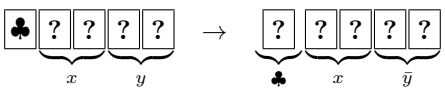
このとき，$x \wedge y = 1$ のときに限り，黒のカード 3 枚が巡回的に連続しないことが確認できる．

(2) ランダムカットをカード列に適用する．

(3) 5 枚のカード全てを表にすると，3 枚の $\boxed{\clubsuit}$ が巡回的に連続して並ぶか，そうでないかのどちらかになる．前者の場合 $x \wedge y = 0$ であり，後者の場合 $x \wedge y = 1$ である．

黒ベース OR

(1) 追加カードを $\boxed{\clubsuit}$ として赤ベース AND（Five-Card Trick）のステップ 1 を実行する．するとカード列は次のようになる．

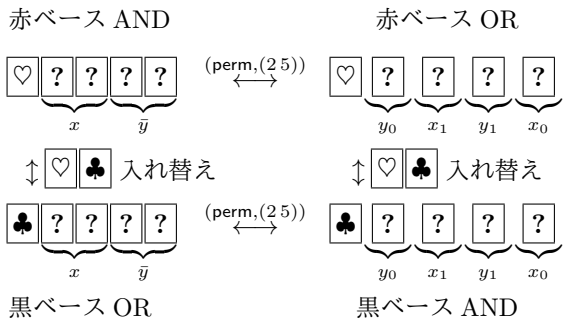


このとき，$x \vee y = 1$ のときに限り，黒のカード 3 枚が巡回的に連続しないことが確認できる．

(2) ランダムカットをカード列に適用する．

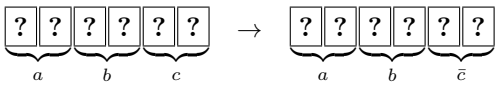
(3) 5 枚のカード全てを表にすると，3 枚の $\boxed{\clubsuit}$ が巡回的に連続して並ぶか，そうでないかのどちらかになる．前者の場合 $x \vee y = 0$ であり，後者の場合 $x \vee y = 1$ である．

4 つのプロトコルには次の関係が成り立つ．



3.2 提案非コミット型プロトコルのアイデア

提案非コミット型プロトコルのアイデアを示す．初期状態は次の通りであり，最初に c のコミットメントの左右のカードを並び替え，$\bar{c}$ のコミットメントとする．



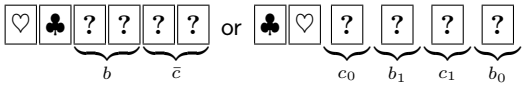
このカード列の 2 枚目から 6 枚目のカードに対してランダムカットを適用すれば，b と c のコミットメントに対して，$a = 0$ のとき赤ベース AND が，$a = 1$ のとき黒ベース OR が適用され，$\mathsf{maj}(a, b, c)$ を計算できる．しかし，出力の際に赤ベースか黒ベースかによって a の値が漏れてしまう．

そこで，上で示した 4 つのプロトコルの関係を利用し，$a = 0$ のとき赤ベース AND もしくは黒ベース AND がそれぞれ 1/2 の確率で適用されるようにランダム化を行う．（自動的に，$a = 1$ のとき黒ベース OR もしくは赤ベース OR がそれぞれ 1/2 の確率で適用される．）具体的には上のカード列に $(\mathsf{shuf}, \{\mathsf{id}, (1\,2)(3\,6)\})$ が適用されるように並び替えとランダム二等分割カットを行う．このとき，カード列の状態は次のようになる．

(1) $a = 0$ の場合

2 枚目以降のカード列にランダムカットを適用することで，前者は赤ベース AND が，後者は黒ベース AND が適用されたことになり $b \wedge c$ が計算できる．

(2) $a = 1$ の場合



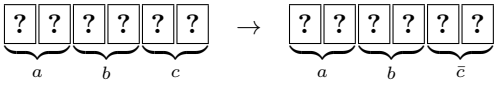
2 枚目以降のカード列にランダムカットを適用することで，前者は黒ベース OR が，後者は赤ベース OR が適用されたことになり $b \vee c$ が計算できる．

結局 4 つのどの場合でも，2 枚目以降のカード列にランダムカットを適用すれば良いため，値が漏れることはない．$a = 0$ のときに $b \wedge c$，$a = 1$ のときに $b \vee c$ を計算できるため，$\mathsf{maj}(a, b, c)$ の値を得ることができる．

3.3 提案非コミット型プロトコルの手順

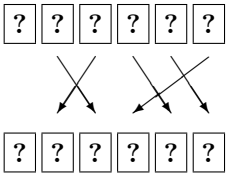
提案非コミット型プロトコルの手順を示す．提案プロトコルは入力値 a, b, c のコミットメントの 6 枚のカードでスタートする．

(1) a, b, c のコミットメントを並べ，c のコミットメントの否定をとる．

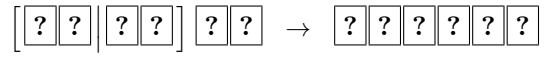


(2) $(\mathsf{shuf}, \{\mathsf{id}, (1\,2)(3\,6)\})$ を次のように適用する．

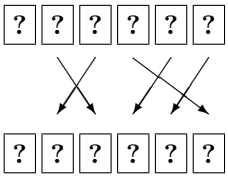
(a) カードを次のように並べかえる．



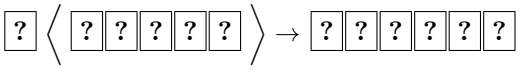
(b) 1 枚目から 4 枚目のカードにランダム二等分割カットを適用する．



(c) カードを次のように並べかえる．

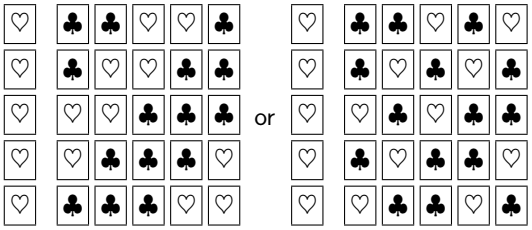


(3) 2 枚目以降のカード列にランダムカットを適用する．

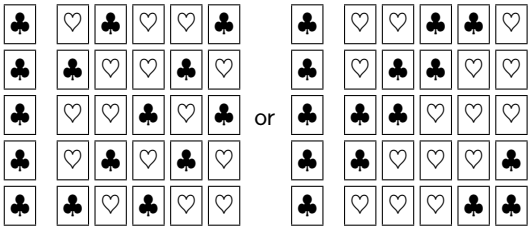


(4) 1 枚目のカードをめくる．

(a) ♡ が出た場合，残りの 5 枚をめくり，♣ が 3 つ連続していれば 0，そうでなければ 1 が出力である．



(b) ♣ が出た場合，残りの 5 枚をめくり，♡ が 3 つ連続していれば 1，そうでなければ 0 が出力である．



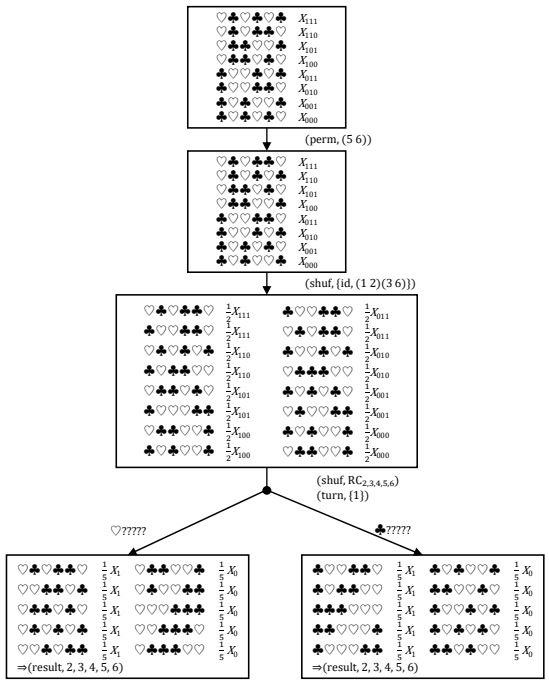


図 1 提案非コミット型プロトコルの KWH-tree.（$\mathsf{shuf}, \mathsf{RC}_{2,3,4,5,6}$）は 2 枚目から 6 枚目のカードにランダムカットを適用することを意味する.

以上が提案非コミット型プロトコルの手順である．このプロトコルはループを含まないため，必ず 2 回のシャッフルで終了する.

3.4 提案非コミット型プロトコルの正当性と安全性

提案非コミット型プロトコルの正当性と安全性を KWH-tree [6] を用いて証明する．KWH-tree とは，カードの状態を表すノードとそれらを結ぶ操作を表すエッジでプロトコルを表現する図である．各ノードの確率分布の和が根の確率分布の和と等しいという条件を満たしつつ KWH-tree を描くことができれば，そのプロトコルの正当性と安全性が証明される．提案非コミット型プロトコルは，図 1 のように KWH-tree が描ける．したがって，このプロトコルは正当かつ安全である.

4. 提案コミット型プロトコル

本節では，最初に非コミット型の提案プロトコルをコミット型に変形するアイデアを述べ，その後手順を説明する．さらに，その正当性と安全性を証明する.

4.1 提案コミット型プロトコルのアイデア

提案するコミット型プロトコルは Abe らが提案した 5 枚 AND プロトコル [5] から着想を得ている．Abe らが提案した 5 枚 AND プロトコルは Five-card Trick に，さらに操作を加えることによって 2 枚のカード，すなわちコミットメントで出力することを実現している．このアイデアを利用し，提案非コミット型プロトコルの最後の部分で残りの 5 枚のカードをめくらずにコミットメントで出力できるように操作を加えることでコミット型への変形を実現する.

以下に Abe らのプロトコルのアイデアを示す.

(1) Five-Card Trick のステップ 1〜3 を行う．ここで，中央のカードをめくり，$\boxed{\clubsuit}$ が出たと仮定する.

$\boxed{?}\boxed{?}\boxed{\clubsuit}\boxed{?}\boxed{?}$

このとき，カード列の状態は次の 4 通りのいずれかである.

(i)	♡♡♣♡♣	$a \wedge b = 0$
(ii)	♣♡♣♡♡	$a \wedge b = 0$
(iii)	♡♡♣♣♡	$a \wedge b = 1$
(iv)	♡♣♣♡♡	$a \wedge b = 1$

中央のカードを伏せた後，説明のために次のようにカードに記号をつける.

$\underbrace{\boxed{?}\boxed{?}}_{x}\underbrace{\boxed{?}\boxed{?}}_{y}\boxed{?}$

このとき，(ii) と (iv) の場合は x と y それぞれのカード組をコミットメントとみなすことができ，$x \oplus y = a \wedge b$ となっている．したがって，これらの場合は左側 4 枚のカードに XOR プロトコル [9] を適用することで，$x \oplus y = a \wedge b$ のコミットメントを得ることができる．もし，(i) または (iii) の場合であっても，XOR プロトコルを適用することで情報が漏れることはない.

以上が Abe らのプロトコルのアイデアである．このアイデアを用いた提案するコミット型プロトコルの詳しい手順を次節で示す.

4.2 提案コミット型プロトコルの手順

提案するコミット型プロトコルの手順を説明する．ステップ 2 でカードをめくる操作を行うが，$\boxed{\clubsuit}$ が出た場合と $\boxed{\heartsuit}$ が出た場合の操作は対称性があるため，ここでは $\boxed{\clubsuit}$ が出た場合を説明する．（$\boxed{\heartsuit}$ が出た場合については $\boxed{\clubsuit}$ と $\boxed{\heartsuit}$ を入れ替えて考えればよい.）

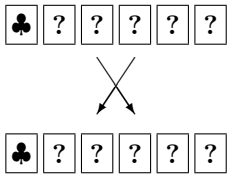
(1) 非コミット型プロトコルのステップ 1〜3 を行う.

(2) 1 枚目のカードをめくる．（ここでは $\boxed{\clubsuit}$ が出た場合を説明する.）

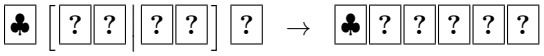
(3) 4 枚目のカードをめくる．$\boxed{\heartsuit}$ が出た場合，そのカードを裏にし，2 枚目から 6 枚目のカードにランダムカットを適用してからこのステップをやり直す．$\boxed{\clubsuit}$ が出た場合，そのカードを裏にして次のステップへ進む.

(4) 2 枚目から 5 枚目のカードに対し，XOR プロトコルを適用する.

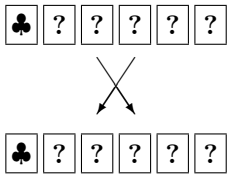
(a) 次のように並べ替える.



(b) ランダム二等分割カットをカード列に適用する．

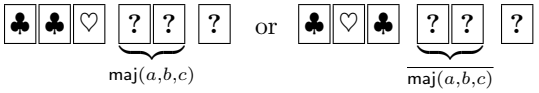


(c) 次のように並べ替える．



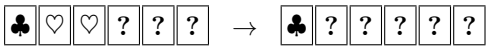
(5) 2 枚と 3 枚目のカードをめくる．

(a) ♣♡ あるいは ♡♣ が出たら，maj(a,b,c) のコミットメントが得られる．

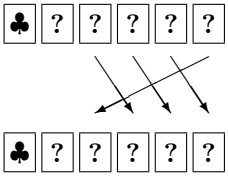


後者の場合は，左右のカードを入れ替えることで，maj(a,b,c) のコミットメントが得られる．

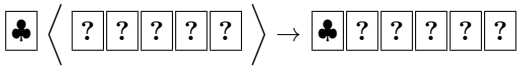
(b) ♡♡ が出たら，そのカードを裏にして，



次のように並べかえる．



その後，ランダムカットを適用してステップ 3 に戻る．



以上が提案するコミット型プロトコルの詳しい手順である．このプロトコルは期待値として 9 回のシャッフルで終了する．

4.3 提案コミット型プロトコルの正当性と安全性

提案するコミット型プロトコルの正当性と安全性を 3.4 節と同様に，KWH-tree を用いて証明する．このプロトコルは，図 2 のような KWH-tree で描ける．したがって，このプロトコルは正当かつ安全である．

5. おわりに

本論文では，非コミット型の 3 入力多数決関数プロトコルを背面処理を用いずに 6 枚のカードで構成した．さらに，非コミット型からコミット型へ変形する方法について示した．

本研究によって，コミット型 3 入力多数決関数プロトコルの構成に必要なカード枚数は 6 枚であることが判明した．有限時間かつコミット型については Nishida ら [1] が示した 8 枚のプロトコルしか存在しないため，その枚数の必要十分条件を解明することが今後の方針である．

参考文献

[1] T. Nishida, T. Mizuki, and H. Sone, “Securely computing the three-input majority function with eight cards,” TPNC 2013, Vol. LNCS 8273, pp.193–204, 2013.

[2] T. Nakai, S. Shirouchi, M. Iwamoto, and K. Ohta, “Four cards are sufficient for a card-based three-input voting protocol utilizing private permutations,” ICITS 2017, Proceedings, volume 10681 of Lecture Notes in Computer Science, pp.153–165, Springer, 2017.

[3] Y. Watanabe, Y. Kuroki, S. Suzuki, Y. Koga, M. Iwamoto, and K. Ohta, “Card-based majority voting protocols with three inputs using three cards,” In International Symposium on Information Theory and Its Applications, ISITA 2018, pp.218–222. IEEE, 2018.

[4] K. Yasunaga, “Practical Card-Based Protocol for Three-Input Majority,” IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, 2020.

[5] Y. Abe, Y. Hayashi, T. Mizuki and H. Sone, “Five-Card AND Protocol in Committed Format Using Only Practical Shuffles,” Proceedings of the 5th ACMon ASIA Public-Key Cryptography Workshop (APKC 2018), pp.3–8, 2018.

[6] A. Koch, S. Walzer, and K. Härtel, “Card-based cryptographic protocols using a minimal number of cards,” ASIACRYPT 2015, Lecture Notes in Computer Science, Vol. 9452, pp.783-807, 2015.

[7] T. Mizuki and H. Shizuya, “A formalization of card-based cryptographic protocols via abstract machine,” International Journal of Information Security, vol.13, no.1, pp.15–23, 2014.

[8] I. Ueda, A. Nishimura, Y. Hayashi, T. Mizuki and H. Sone, “How to Implement a Random Bisection Cut,” Theory and Practice of Natural Computing, Lecture Notes in Computer Science, Vol. 10071, pp.58-69, 2016.

[9] T. Mizuki and H. Sone, “Six-card secure AND and four-card secure XOR,” Fron-tiers in Algorithmics, eds. by X. Deng, J.E. Hopcroft, and J. Xue, vol.5598, pp.358–369, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2009.

[10] I. Ueda, D. Miyahara, A. Nishimura, Y. Hayashi, T. Mizuki and H. Sone, “Secure implementations of a random bisection cut,” International Journal of Information Security, pp.445-452, 2020.

[11] B. den Boer, “More efficient match-making and satisfiability the five card trick,” EUROCRYPT 1989, Vol. LNCS 434, pp. 208–217, 1990.

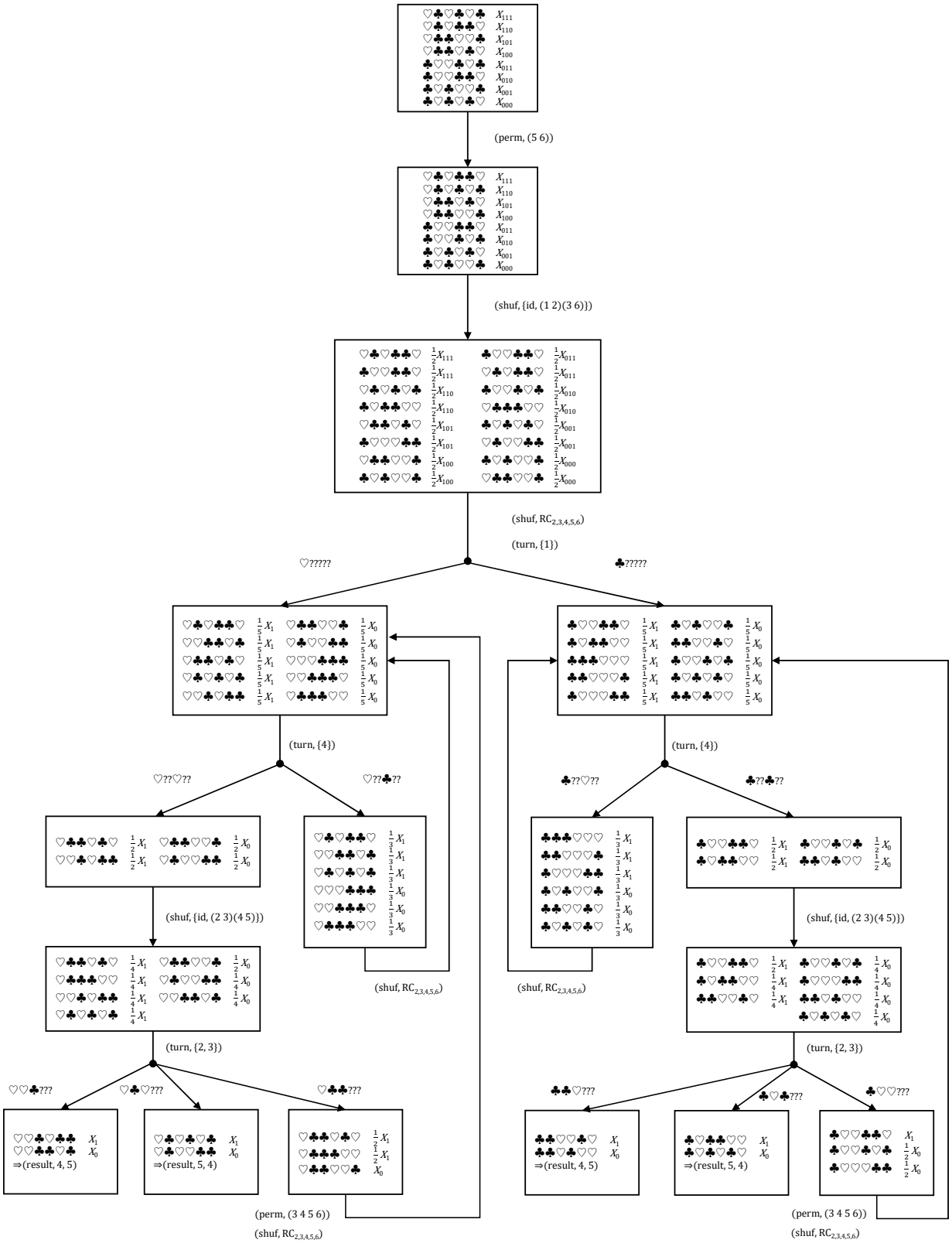


図 **2**　提案コミット型プロトコルの KWH-tree