Vol.2022-DPS-190 No.36
Vol.2022-CSEC-96 No.36
2022/3/11

3値入力可能な拡張 Five Card Trick における第4の未定義値の扱いについて

須賀 祐治[1,a)]

概要：2 者間の AND 演算によるマッチングはカードベースプロトコルにおける一般的なアプリケーションであり，気まずくならない告白ができることが知られている．2 者間の秘密計算によって AND 演算出力が 0 である場合，入力が 0 だったのか 1 だったのかを秘匿できる意味で，相手に入力がバレないことから気まずくならないとされている．本稿は 0,1 という 2 択の入力を持つ通常の AND 演算を拡張し，0 でも 1 でもない第 3 の値「不定」を入力可能な拡張 AND プロトコルを考える．このカードプロトコルでは一般的なエンコードに基づく方式ではなく，カードの回転により裏面の識別不可能性が失われない同一カードを用いる．SCIS2022 にて Five Card Trick とほぼ同様の操作で $0, 1, \theta$ を入力可能なプロトコルが実際に示されており，位数 3 の半群の条件を満たす 3 値論理となるような代数的構造を持つように構成されており非コミットメント型からコミットメント型への移行がスムーズになるような設計が行われている．
本稿は，上記のように拡張された Five Card Trick において，第 4 の未定義値（不正値）が入力された場合の考察を行う．まず SCIS2022 での提案方式に対して未定義値入力が検知可能かについて触れる．次に根本的な対策として第 4 値の入力を防ぐ方法として，同一カードに加えシール貼付を行う方法を提案する．上下シャッフルによって裏面の識別不可能性を維持できる例としては，片面印刷の名刺などが利用でき，準備も操作も簡便な方式であることから現実的な方式であると考えられる．

キーワード：Card-based protocols, Non-committed protocols, Five Card Trick, Commutative semigroups

Security considerations for the fourth data over non-commitment 3-input extended Five Card Trick card-based protocols

Yuji Suga[1,a)]

Abstract: The matching situation with AND operations between two parties is a common application in card-based protocols, and it is known to provide a non-embarrassing confession of love. This means that the other party does not know whether the input was 0 or 1, which is said to avoid embarrassment. In this paper, we consider an extended AND protocol that allows the input of a third value "indefinite", which is neither 0 nor 1, by extending the normal AND operation with two input choices, 0 and 1. In SCIS2022, protocols that allow the input of $0, 1 and \theta$ with almost the same operation as the Five Card Trick has been actually proposed. The protocols are designed to have an algebraic structure such that it is a 3-valued logic satisfying the condition of a semigroup with 3 elements, and the transition from uncommitted to committed protocols is possible.
In this paper, we consider the case where the fourth undefined value (illegal value) is input to the Five Card Trick extended as described above. First, we discuss whether the proposed method in SCIS2022 can detect the input of the undefined value. Next, as a fundamental countermeasure to prevent the input of the fourth value, we propose a method of attaching stickers to the same cards. As an example of a method that can maintain the indistinguishability of the reverse side by shuffling the cards up and down, business cards printed on one side can be used, which is a practical method because it is easy to prepare and shuffle.

Vol.2022-DPS-190 No.36
Vol.2022-CSEC-96 No.36
2022/3/11

1. はじめに

一般的なカードベースプロトコルでは，入力を行う各プレイヤーは semi-honest であると仮定されていることからプレイヤーは不正できないことが前提となっている．これは傍観者がいる状況でも同様であるが，この京菜シチュエーションにおいて不正な操作を行う場合の問題点が指摘されており，物理的なカードの製造に起因する秘密漏えい等も含めて攻撃対策方法が提案されている [15]．また，プロトコルによってはプレーヤーの背面操作（Private operaions とも呼ばれる）が必要な種類のプロトコルが存在しており，プレイヤーの不正に対する検知や根本的な対策についても研究されている [17] [18] [20] [21]．さらに不正行為が分かった上での攻撃のモデルも提案されており，非コミットメント型プロトコルにおいて不正にカードを開示することで入力を得ようとする不正開示攻撃の対策も考えられている [16] ユーザビリティの観点での研究も進められており，カードの並べ替え [22], コインベースプロトコルにおける初期入力誤り [23] に関する考察も行われている．

本稿では背面操作を特に必要としない一般的なカードベースプロトコルにおいて，入力時に規定値以外にも余地のあるエンコーディングルールを用いる際の不正入力に関して考察する．特に SCIS2022 で発表された 3 値入力可能な Five Card Trick [30] において，入力エンコーディングの規定に無い不正入力の検知について検討する．

1.1 本稿で扱う対象

本稿では表面裏面ともに全く同じ絵柄であるカード（例えば名刺や麻雀牌）を用いることを考える（文献 [2] において水木らは絵柄の上下関係を生かしたメリットとデメリットについて考察されている）．このときカードの上下配置の違いを用いて，それぞれ（一般的なカードプロトコルで用いられる）異なるスートと対応づけることができる．つまり $\boxed{\downarrow}$ を $\boxed{\clubsuit}$ と，$\boxed{\uparrow}$ を $\boxed{\heartsuit}$ と同一視することを考える．スートを表現するメモ書きをしないでも，同一カードの束を用いてプロトコルを構成することができる点も一つのメリットである．

2. 非コミットメント型プロトコル

本稿はカードベースプロトコルのうち非コミット型のプロトコルを扱う．一般的なカードベース暗号では 1 ビット入力を 2 種類 2 枚のカードが用いられる [1]．例えば，ユーザによる 1 ビット入力は以下の一般的なエンコーディングルールに従う：$\boxed{\clubsuit}\boxed{\heartsuit} = 0$, $\boxed{\heartsuit}\boxed{\clubsuit} = 1$.

出力がコミット型であるとは，プロトコル停止時に得られる結果が，入力のエンコーディングルールに基づいた形式であることを指す．一方で非コミット型であるとは，プロトコル停止時に利用されたカードを開示するなどして結果を得る方式である．

2.1 オリジナル Five-Card Trick

2 ユーザによる非コミットメント型として知られる Five-card trick [3] はハートとクラブ 2 種類のカードが用いられている．

Five-card trick は 2 ユーザ間で AND 演算を行うプロトコルである．2 入力を $a, b \in \{0, 1\}$ としたとき $\boxed{?}\boxed{?}(= \bar{a})$ $\boxed{\heartsuit}$ $\boxed{?}\boxed{?}(= b)$ として 5 枚のカードを並べてランダムカット（巡回置換を c_5 としたとき，恒等置換 id と c_5, c_5^2, c_5^3, c_5^4 の 5 通りから等確率で選択してカード束に処理する操作）を行う．ここで $\boxed{?}$ は裏面にして入力したことを示しており $\bar{a}$ は a の否定 (negation) である．

ランダムカットを行う際には中央の $\boxed{\heartsuit}$ も $\boxed{?}$ とし，5 枚とも裏面に向けてシャッフルする．出力は 5 枚のカードをすべて開示することで得られる．3 枚の $\boxed{\heartsuit}$ が連続して並んで出力されたとき $a \wedge b = 1$，それ以外は $a \wedge b = 0$ となる．以下は 5 枚のカードの初期状態を示しており，これらの 5 枚のカードが巡回置換によりシャッフルされることから，出力時に 3 枚の $\boxed{\heartsuit}$ が連続して並んでいる場合のみが $a \wedge b = 1$ となることが分かる．さらに，$a \wedge b = 0$ となる 3 つのケースについてはすべて同一視されるため，出力だけを見ても入力 a, b がどのような値だったかについて認識できない点が本プロトコルのポイントである．

(a, b)	sequence
(0,0)	♡ ♣ ♡ ♣ ♡
(0,1)	♡ ♣ ♡ ♡ ♣
(1,0)	♣ ♡ ♡ ♣ ♡
(1,1)	♣ ♡ ♡ ♡ ♣

表 1 Five-Card Trick 初期入力状態

Five-Card Trick は，カード入力時の一般置換（このケースでは b を入力の際にカードを倒置して置いているため 5 枚のカード全体として $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 5 & 4 \end{pmatrix}$ の置換を行っていると考える）とランダムカットのみで構成されるシンプルなプロトコルである．

2.2 上下シャッフルの導入と Three Card Trick

Five Card Trick と同様に 2 者の AND 演算プロトコルを考える．一般的なカードプロトコルにおいては 2 種類のスート各 1 枚の計 2 枚が配布され，2 枚のカードで 1 ビッ

[1] 株式会社インターネットイニシアティブ
Internet Initiative Japan Inc., Iidabashi Grand Bloom, 2-10-2 Fujimi, Chiyoda-ku, Tokyo 102-0071, Japan
[a)] suga@iij.ad.jp

トを表現する．そのため配布される最小枚数は 2 枚である．一方で，表面裏面ともに全く同じ絵柄であるカードを用いる場合には 1 枚のカードを上下関係，つまり 2 種類の方向 $\boxed{\downarrow}\boxed{\uparrow}$ で入力可能なことから 1 ビットを 1 枚で表現可能である．そのため配布される最小枚数は 1 枚であり，1 枚のカードを配布した際の AND プロトコルが構成できれば，枚数としては optimal な方式であると言える．

例えば $\boxed{\downarrow}=0$, $\boxed{\uparrow}=1$ というエンコーディングルールを適用しようとする場合 5 Card Trick のような入力，つまり真ん中のエクストラカードを $\boxed{\uparrow}$ とし，両側から各ユーザが a,b 1 枚ずつの裏面入力を行うと初期状態として以下のような配置となる．仮にこれを Three Card Trick と呼ぶこととする．

(a,b)	sequence
(0,0)	$\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}$
(0,1)	$\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}$
(1,0)	$\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}$
(1,1)	$\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}$

表 2 Three-Card Trick 初期入力状態

Five Card Trick と同じように入力を撹乱するため 3 枚カードをランダムカットのカード処理を行うが，出力時には $a\wedge b=1$ のときのみ エクストラカードを含めて $\boxed{\uparrow}$ が 3 枚並んでいることが分かる．一方で Five Card Trick において $a\wedge b=0$ となる 3 つのケースはすべて同一視できていたが表 2 のように $\boxed{\uparrow}$ となるカードの枚数が異なることから同一視できず，このままでは入力 a,b を秘匿できない．

そのため次のテクニックを用いる．アイデアとしてはランダム 2 等分カット [5] を勘弁に実装する方式で用いられている方式によく似通っており，上下関係をランダムに入れ替えること（この操作を上下シャッフルと呼ぶ）を考える．ひとつの方法としてはランダムカット後のカード束にさらにエクストラカード 1 枚を用いて 3 枚のカードのうちの 1 枚めの表面を秘匿し，放り投げる等して上下関係を入れ替えた後にエクストラカードを抜き去るという方式が考えられる．ここは様々な実装方式があると考えるが，いずれにせよ上下シャッフルは $\boxed{\downarrow}$ と $\boxed{\uparrow}$ が入れ替わることを意味しており Three Card Trick では以下のように初期入力状態が変化することとなる．

(a,b)	sequence
(0,0)	$\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}$
(0,1)	$\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}$
(1,0)	$\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}$
(1,1)	$\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}$

表 3 Three-Card Trick にて上下シャッフル後の初期配置

表 2 と 表 3 の状態をすべて見比べたとき，ランダムカット→上下シャッフル後の状態は $\boxed{\uparrow}$ となるカードが 0,1,2,3 枚のいずれかとなり，$a\wedge b=1$ のときは $\boxed{\uparrow}$ が 0,3 枚のいずれかである．さらに $a\wedge b=0$ のときは $\boxed{\uparrow}$ が 1,2 枚であり，かつ 入力 a,b が秘匿された状態で出力を得ることができる．つまり $\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}$, $\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}$, $\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}$, $\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}$ の 6 通りはすべて同一視されることから本プロトコルの安全性（入力の秘匿性）を確保していることが分かる．また，Three Card Trick は入力カード枚数として optimal な AND 演算プロトコルを実現していることが分かる．

同一視できる 3 枚組に関して以下のタイプに分類することができる．

タイプ	同一視されるカード組
1	$\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}$, $\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}$, $\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}$, $\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}$
3	$\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}$, $\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}$

表 4 ランダムカット→上下シャッフル後の分類（3 枚組）

タイプ名は何枚 $\boxed{\uparrow}$ カードが連続するか最大の値を示す．例えばタイプ 3 にカテゴライズされる $\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}$ がそれに該当し，上下シャッフルにより $\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}$ と同一視されることも分かる．

3. 3 値入力と 3 値論理

Five-Card Trick に限らずカードベースプロトコルの多くは 2 値 True(1), False(0) を入力することが想定されている．例えば 2 人による AND プロトコルは「気まずくならない告白」ができるとされており，実際プレーヤー A が True を入力して False という結果を得たとしても，相手のプレイヤー B には A が True を入力したことがバレることがない，という側面で気まずくならない点を保証している．

このようなシチュエーションを考えた場合，果たしてプレイヤーは 0 か 1 の 2 つの選択しか認めないのであろうか．相手に好意を寄せているのかどうかは 2 値ではなく，中間的な気持ちである「どちらでもない」「自分の気持ちが分からない」を排除してよいのか，という課題を考える [29]．そこでカードプロトコルにおいて 0 でも 1 でもない第 3 の値を入力できるようにすることが可能にする．

3.1 3 値論理のバリエーション

第 3 の値を入力するとして 入力 a,b に対して $a\wedge b$ の真偽表としてどれを用いるかという問題を考える．従来の 2 値論理における AND 真偽表は以下となる．

$a\backslash b$	0	1
0	0	0
1	0	1

これに対して第 3 の値 θ を入れた際に様々な考え方が存在する．一例として，より代数的な考慮に基づいた Lukasiewicz の 3 値論理における AND 真偽表は以下となる．

$a\backslash b$	0	θ	1
0	0	0	0
θ	0	θ	θ
1	0	θ	1

0, 1 をそれぞれ体におけるゼロ元，単位元として考えたときのストレートフォワードな方式である．特に $\theta^2 = \theta$ を満たすように構成されている点に留意する．Kleene による 3 値論理においても AND 演算に関しては同じ真偽表を持つことが知られている．

次に Bochvar の 3 値論理における AND 真偽表を示す．

$a\backslash b$	0	θ	1
0	0	θ	0
θ	θ	θ	θ
1	0	θ	1

既に取り上げたように，この類のプロトコルでは「気まずくならない告白」ができるが，さらに「気持ちが揺らいでいる」ことも分かる，という観点で，ゲーム理論の側面としても面白い構造を持っていることが分かる．具体的には a または b のどちらかが θ を入力しただけで AND 演算の結果が θ となる点が面白い．しかし，これは秘密計算としては条件をバイオレーションしており，例えばプレイヤー A の入力が 0 または 1 のときには，マッチングがうまくいかない場合，プレイヤー B にその入力を秘匿することができるが，θ を入力してしまうと，入力が漏れてしまう．本稿ではこれをセキュリティ要件を満たさないという立場ではなく，このリスクを許容してプレイヤーは入力することを前提とする，という立ち位置で議論していく．

3.2 位数 3 の可換半群

ビルディングブロックとして 2 者間の拡張 AND 演算プロトコルを用い，複数のプロトコルを連続して実行することを想定すると，この拡張演算は推移律を満たす必要がある．さらに AND 演算は可換であることから，位数 3 の可換半群がここで扱うべき対象となる．計算機による数え上げではなく手による証明を行い，自明な場合を取り除いた位数 3 の可換半群のうち AND の代数構造を持つ半群の真偽表は 7 種類となる [30] [31]．本稿では代数的に同型で既に 5 枚を利用したカードプロトコルが実現可能である以下の 3 つのパターンについてのみ扱う．

AND-$(0,\theta,\theta)$:

$a\backslash b$	0	θ	1
0	0	0	0
θ	0	θ	θ
1	0	θ	1

AND-$(\theta,0,\theta)$:

$a\backslash b$	0	θ	1
0	0	θ	0
θ	θ	0	θ
1	0	θ	1

AND-$(0,0,\theta)$:

$a\backslash b$	0	θ	1
0	0	0	0
θ	0	0	θ
1	0	θ	1

上記 AND-$(0,\theta,\theta)$, AND-$(\theta,0,\theta)$, AND-$(0,0,\theta)$ の 3 方式について optimal と考えられる方式について説明する．

3.3 AND-$(0,\theta,\theta)$ の実装

エンコーディングルールとして以下を適用する：$\boxed{\downarrow}\boxed{\uparrow}= 0$, $\boxed{\uparrow}\boxed{\downarrow}= \theta$, $\boxed{\downarrow}\boxed{\downarrow}= 1$．このとき真ん中にエクストラカード $\boxed{\uparrow}$を置いてその左側に a の negation, 右側に b を入力した場合，バリエーションとして表 5 の 9 パターンが得られる．ここで negation は左右を入れ替えたカードを入力することとする．Five Card Trick では 補数の入力を意味していたが，Five Card Trick においても negation は 2 枚のカードの左右を入れ替える操作と考えれば全く同じ操作であると考えることができる．

(a,b)	sequence
(0,0)	$\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}$
(0,1)	$\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}$
(1,0)	$\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}$
(1,1)	$\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}$
(0,θ)	$\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}$
(θ,0)	$\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}$
(1,θ)	$\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}$
(θ,1)	$\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}$
(θ,θ)	$\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}$

表 5 AND-$(0,\theta,\theta)$ 実装の初期状態

このときランダムカットと上下シャッフルと行うことにより $(a,b) = (1,1)$ のときのみ $\boxed{\uparrow}$が 1 または 4 枚のパターンが現れる．また $(a,b) = (\theta,\theta),(\theta,1),(1,\theta)$ の場合

$\boxed{\uparrow}$または $\boxed{\downarrow}$が 3 枚連続現れ，これらの 3 パターンは全て同一視される．さらにそれ以外の 5 パターンは例えば $\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}$等が現れ，この形式がランダムカットと上下シャッフルした状態のいずれかに一致するため同一視される．

このことから注意深く分類すると

$a\backslash b$	0	θ	1
0	0	0	0
θ	0	θ	θ
1	0	θ	1

という真偽表が得られることから Five Card Trick に上下シャッフル操作を追加するだけで AND-$(0,\theta,\theta)$ を実装できることが分かる．

表 4 と同様に，ランダムカット→上下シャッフル後の分類を行うと以下となる．

タイプ	同一視されるカード組
2	$\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}$, $\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}$ $\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}$ $\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}$, $\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}$, $\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}$ $\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}$, $\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}$
3	$\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}$, $\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}$, $\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}$ $\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}$, $\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}$ $\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}$, $\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}$ $\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}$, $\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}$
4	$\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}$, $\boxed{\downarrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}$ $\boxed{\downarrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}$, $\boxed{\uparrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}$ $\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}$, $\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\uparrow}\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}$ $\boxed{\uparrow}\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}$, $\boxed{\downarrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}$

表 6 ランダムカット→上下シャッフル後の分類（5 枚組）

タイプ 5 に分類される$\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}\boxed{\uparrow}$, $\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}\boxed{\downarrow}$は出現せず，上記のように $2^5-2=30$ 通りが出現することとなる．

さきほどの真偽表にて上記タイプ (3,4,5 のいずれか) を当てはめると以下となる．ただし a の入力としては negation を表記している，つまり実際のカード入力であることに注意する．

$\bar{a}\backslash b$	$\boxed{\downarrow}\boxed{\uparrow}$	$\boxed{\uparrow}\boxed{\downarrow}$	$\boxed{\downarrow}\boxed{\downarrow}$
$\boxed{\uparrow}\boxed{\downarrow}$	Type-2	Type-2	Type-2
$\boxed{\downarrow}\boxed{\uparrow}$	Type-2	Type-3	Type-3
$\boxed{\downarrow}\boxed{\downarrow}$	Type-2	Type-3	Type-4

表 7 AND-$(0,\theta,\theta)$ 実装のタイプ分類

出現として Type-2 が 出力 0, Type-3 が出力 θ, Type-4 が出力 1 に該当することが分かる．

4. 第 4 値入力時の考察

前章で説明した AND-$(0,\theta,\theta)$ の実装においては入力として$\boxed{\downarrow}\boxed{\uparrow}=0$, $\boxed{\uparrow}\boxed{\downarrow}=\theta$, $\boxed{\downarrow}\boxed{\downarrow}=1$ のみが規定されており$\boxed{\uparrow}\boxed{\uparrow}$の入力を想定していないことが分かる．そこで，表 7 に倣い，もし第 4 の値$\boxed{\uparrow}\boxed{\uparrow}$が入力された場合の真偽表を考える．

$\bar{a}\backslash b$	$\boxed{\downarrow}\boxed{\uparrow}$	$\boxed{\uparrow}\boxed{\downarrow}$	$\boxed{\downarrow}\boxed{\downarrow}$	$\boxed{\uparrow}\boxed{\uparrow}$
$\boxed{\uparrow}\boxed{\downarrow}$	Type-2	Type-2	Type-2	**Type-4**
$\boxed{\downarrow}\boxed{\uparrow}$	Type-2	Type-3	Type-3	**Type-4**
$\boxed{\downarrow}\boxed{\downarrow}$	Type-2	Type-3	Type-4	**Type-3**
$\boxed{\uparrow}\boxed{\uparrow}$	**Type-4**	**Type-4**	**Type-3**	**Type-5**

表 8 AND-$(0,\theta,\theta)$ 実装のタイプ分類

結果として 入力 a,b ともに不正の第 4 値である $\boxed{\uparrow}\boxed{\uparrow}$を入力したのみ Type-5 が出現することから第 3 者（傍観者）が検知可能であることが分かる．それ以外のケースについては入力が $\boxed{\downarrow}\boxed{\uparrow}$または $\boxed{\uparrow}\boxed{\downarrow}$についてはありえない出力になることから検知が可能である．例えば $\bar{a}$ として $\boxed{\uparrow}\boxed{\downarrow}$が入力された場合，Type-2 か Type-3 しか出現されないはずであるが， b として 不正値である $\boxed{\uparrow}\boxed{\uparrow}$が入力されると Type-4 が出現することから検知可能である．一方で $\bar{a}$ として $\boxed{\downarrow}\boxed{\downarrow}$が入力されて Type-3 が出力として出現した場合，正当な入力なのか不正値なのか区別することができない．よって，この方式は入力を制限して利用することが望まれることとなる．もしくは根本的に $\boxed{\uparrow}\boxed{\uparrow}$が入力できないような実装方法を検討する必要があることが分かる．

4.1 $\boxed{\uparrow}\boxed{\uparrow}$の入力を制限する根本的対策

プレイヤーは 2 枚のカードを持つことから $\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\uparrow}\boxed{\downarrow}$, $\boxed{\downarrow}\boxed{\downarrow}$, $\boxed{\uparrow}\boxed{\uparrow}$の 4 パターンが入力できるが，先に述べたように $\boxed{\uparrow}\boxed{\uparrow}$を入力することでプロトコルが破綻してしまう．そのため $\boxed{\uparrow}\boxed{\uparrow}$が入力できない根本的解決が必要となる．

これまでカードとして裏面が識別不可能性を持つ同一カードを利用し上下の入力の違いによりプロトコルへの入力を行う方法について述べた．ここで，同一カードだけでなく同一のデザインを持つシールを併用することを考える．ここでは 5 枚のカードと 10 枚のシールを想定し，シール貼った状態の 5 枚のカードは同一視できるものとする．また，カードからシールを剥がした後はその痕跡を検知できないものとする．カードに 100 円ショップや文具店等で容易に入手可能なラミネート加工フィルムを使用するなどすることでシールを剥がしやすくし，痕跡を残さないようにできると考えられる．

以下プロトコルの実装を説明する．まず識別不可能な状態で 5 枚のカードにそれぞれ 1 枚ずつ裏面ではなく表面

（おもてめん）にシールを貼る．このときシール位置は上下シャッフルによっても同一視できるように配慮が必要である．5 枚のカードは裏面の識別不可能性が同様に必要であり，表面の柄も上下処理が識別不可能性を持つようなデザインである必要がある．シールも同様に上下の識別不可能性が必要となる．シールを 1 枚のカードを 2 枚同じ位置に貼り付けるが，1 枚なのか 2 枚なのかが分かるようにする必要がある．また，カードが逆さまになった場合もシールデザインが変わらないようにする必要がある．

シールを貼った状態の 4 枚のカードをそれぞれ 2 枚ずつ各プレーヤーに配布し，$\boxed{\downarrow}$は 1 枚のシールを剥がす，$\boxed{\uparrow}$は 2 枚のシールを貼ったままとして取り扱うものとする．拡張 Five Card Trick で用いられる 5 枚目のエクストラカードは前章の実装と同じく $\boxed{\uparrow}$のカードでありシールは 2 枚とも貼られたままのカードである．

このとき 各プレイヤーはプロトコル入力の際に，カードの 1 枚もしくは 2 枚のシールを剥がす作業を行う．カードから 2 枚のシールを剥がしてはならず 1 枚は必ず残す必要がある．そしてカード入力と同時にシールの 1 枚を提示する（2 枚のシール剥がす場合には 1 枚を手元などに隠す）．入力時のカードは上下の配慮する必要がない（ただし表面の識別不可能性が必要である）．

これらの手順により，入力は$\boxed{\downarrow}\boxed{\uparrow}$, $\boxed{\uparrow}\boxed{\downarrow}$, $\boxed{\downarrow}\boxed{\downarrow}$の 3 通りに制限される．また第 4 の値である $\boxed{\uparrow}\boxed{\uparrow}$は剥がしたシールの存在により入力できないことを意味する．実際にこれを実装する際には，手元でシールを剥がす時間を 2 枚分以上十分に確保することと，剥がす作業を行っている際には手元を隠す必要があることに留意する．

5. まとめと今後について

SCIS2022 で提案された 3 値入力が可能な拡張 Five Card Trick において，第 4 の未定義値（不正値）が入力された場合の考察を行った．まず SCIS2022 での提案方式に対して未定義値入力が検知可能かについて，真偽表にタイプという「連続する $\boxed{\uparrow}$の最大枚数」の概念を導入して考察し第 4 値の検知の可能性について触れた．次に根本的な対策として第 4 値の入力を防ぐ方法として，同一カードに加えシール貼付を行う方法を提案した．上下シャッフルによって裏面の識別不可能性を維持できる例としては名刺などが利用でき，準備も操作も簡便な方式であることから現実的な方式であると考えられる．今後，ユーザビリティの観点からより簡便な方法について検討を行うとともに，カード枚数だけではなく他の指標についても検討を行う予定である．

参考文献

[1] 水木, 電子情報通信学会 基礎・境界ソサイエティ Fundamentals Review, 2016 年 9 巻 3 号 pp.179-187, カード組を用いた秘密計算, https://www.jstage.jst.go.jp/article/essfr/9/3/9_179/_article/-char/ja
[2] T. Mizuki, H.Shizuya, Practical Card-Based Cryptography, FUN2014, pp.313-324, 2014.
[3] B. denBoer, More efficient match-making and satisability: the five card trick, EUROCRYPT'89, pp.208-217, 1989.
[4] T. Mizuki and H. Sone, Six-card secure AND and four-card secure XOR, International Workshop on Frontiers in Algorithmics, pp.358-369, 2009.
[5] T. Mizuki, M. Kumamoto and H. Sone, The Five-Card Trick Can Be Done with Four Cards, Asiacrypt2012.
[6] T. Nishida, Y. Hayashi, T. Mizuki and H. Sone, Card-based protocols for any boolean function, TAMC2015.
[7] I. Ueda, A. Nishimura, Y. Hayashi, T. Mizuki and H. Sone, How to implement a random bisection cut, The 5th International Conference on Theory and Practice of Natural Computing (TPNC2016), pp.58-69, 2016.
[8] 駒野, コインを用いる新たなマルチパーティ計算, DICOMO2018.
[9] K. Shinagawa, K. Nuida, T. Nishide, G. Hanaoka and E. Okamoto, Size-Hiding Computation for Multiple Parties, ASIACRYPT2016, 2016.
[10] S. Ruangwises, T. Itoh, Securely Computing the n-Variable Equality Function with 2n Cards, TAMC2020, 2020
[11] S. Ruangwises, T. Itoh, Physical Zero-Knowledge Proof for Numberlink Puzzle and k Vertex-Disjoint Paths Problem, New Generation Computing, 2020.
[12] T. Sasaki, D. Miyahara, T. Mizuki, H. Sone, Efficient card-based zero-knowledge proof for Sudoku, Theoretical Computer Science, Volume 839, pp.135-142, November 2020.
[13] Y. Watanabe, Y. Kuroki, S. Suzuki, Y. Koga, M. Iwamoto, K. Ohta, Card-based majority voting protocols with three inputs using three cards, ISITA2018, pp.218–222, 2018.
[14] K. Shinagawa, K. Nuida, A single shufe is enough for secure card-based computation of any boolean circuit, Discrete Applied Mathematics, Vol.289, pp.248–261, 2021.
[15] カードベース暗号プロトコルに対する攻撃に関する考察, 信学技報, vol.113, no.326, ISEC2013-62, pp.21-28, 2013.
[16] 高島, 宮原, 水木, 曽根, 非コミット型カードベースプロトコルと不正開示攻撃の定式化, コンピュータセキュリティシンポジウム 2019(CSS2019), 2F4-3, pp.886-893, 2019.
[17] 安部，山本，岩本，太田，不正検知可能な 3 入力多数決カードプロトコル，3C3-2, SCIS2019.
[18] Y. Abe, M. Iwamoto,K.Ohta, How to Detect Malicious Behaviors in a Card-Based Majority Voting Protocol with Three Inputs, ISITA2020, C01-9, pp.377-381, 2020
[19] H. Ono, Y. Manabe, Card-Based Cryptographic Protocols with the Minimum Number of Rounds Using Private Operations, New Generation Computing, 2020.
[20] H. Ono, Y. Manabe, Card-based cryptographic logical computations using private operations, New Generation Computing, 2021
[21] Y. Manabe, H. Ono, Card-Based Cryptographic Protocols with Malicious Players Using Private Operations, New Generation Computing, 2022
[22] 駒野, 水木, カードベースプロトコルにおける並べ替え誤りに関する考察, 信学技報, vol.117, no.369, ISEC2017-86, pp.95-101, 2017.
[23] 駒野, 水木, コインベースプロトコルの初期配置誤りに関する考察, コンピュータセキュリティシンポジウム 2020(CSS2020), 4D1-5, pp.1283-1288, 2020.
[24] 須賀, 6 カード 3 入力 equality function (Six-Card Trick)

における置換バリエーションの完全分類, 情報処理学会研究報告 Vol.2020-CSEC-91, No.34, 2020.
[25] 須賀, 怠惰なユーザのための非コミット型カードベース暗号, SCIS2021, 2F2-1, 2021.
[26] Y. Suga, Card-based Cryptography Meets Mahjong Tiles, Small-workshop on Communications between Academia and Industry for Security 2021, 2021.
[27] 須賀, 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, 情報処理学会研究報告 Vol.2021-CSEC-92, No.7, 2021.
[28] 須賀, 三人寄ればチーズの知恵, マルチメディア, 分散協調とモバイルシンポジウム 2021(DOCOMO2021), pp.173-178, 2021.
[29] 須賀, 0,1, 不定の 3 値入力が可能な AND 演算カードベースプロトコルの初期検討, 第 44 回情報理論とその応用シンポジウム (SITA2021), 4-3-3, 2021.
[30] 須賀, 3 値入力可能な可換半群の条件を満たす非コミットメント型 AND 演算拡張カードベースプロトコルの構成, SCIS2022, 2F4-2, 2022.
[31] 須賀, 局所的に AND 構造を持つ位数 3 の可換半群の分類証明とカードベースプロトコルへの適用, to be appeared, 2022.