

Card-Based Protocols Using Unequal Division Shuffle*

Akihiro Nishimura Takuya Nishida Yu-ichi Hayashi Takaaki Mizuki

Hideaki Sone


Tohoku University


tm-paper+card5cop[atmark]g-mail.tohoku-university.jp

Abstract

Card-based cryptographic protocols can perform secure computation of Boolean functions. Cheung et al. presented an elegant protocol that securely produces a hidden AND value using five cards; however, it fails with a probability of $1/2$. The protocol uses an unconventional shuffle operation called unequal division shuffle; after a sequence of five cards is divided into a two-card portion and a three-card portion, these two portions are randomly switched. In this paper, we first show that the protocol proposed by Cheung et al. securely produces not only a hidden AND value but also a hidden OR value (with a probability of $1/2$). We then modify their protocol such that, even when it fails, we can still evaluate the AND value. Furthermore, we present two five-card copy protocols using unequal division shuffle. Because the most efficient copy protocol currently known requires six cards, our new protocols improve upon the existing results. We also design a general copy protocol that produces multiple copies using unequal division shuffle.

1 Introduction

Suppose that Alice and Bob have Boolean values $a \in \{0, 1\}$ and $b \in \{0, 1\}$, respectively, each of which describes his/her private opinion (or something similar), and they want to conduct secure AND computation, i.e., they wish to know only the value of $a \wedge b$. In such a situation, a card-based cryptographic protocol is a convenient solution. Many such protocols have already been proposed (Boe90; CK94; NR98; Sti01; MS09; MKS12; CHL13; KWH15), one of which can be selected by them for secure AND computation. For example, if they select the six-card AND protocol (MS09), they can securely produce a hidden value of $a \wedge b$ using six playing cards, e.g., , along with a “random bisection cut.”

In 2013, Cheung et al. presented an elegant protocol that securely produces a hidden AND value using only five cards (); however, it fails with a probability of $1/2$ (CHL13) (we refer to it as *Cheung’s AND protocol* in this paper). The protocol uses an unconventional shuffling operation that we refer to as “unequal division shuffle”; after a

*An earlier version of this study was presented at 4th International Conference on the Theory and Practice of Natural Computing, TPNC 2015, Spain, December 15–16, 2015, and appeared in Proc. TPNC 2015, Lecture Notes in Computer Science, Springer International Publishing, vol. 9477, pp. 109–120, 2015 (NNH⁺15).

sequence of five cards is divided into a two-card portion and a three-card portion, these two portions are randomly switched. The objective of this paper is to improve Cheung's AND protocol and propose other efficient protocols using unequal division shuffle.

This paper begins by presenting some notations related to card-based protocols.

1.1 Preliminary Notations

Throughout this paper, we assume that cards satisfy the following properties.

1. All cards of the same type (black \clubsuit or red \heartsuit) are indistinguishable from one another.
2. Each card has the same pattern $\boxed{?}$ on its back side, and hence, all face-down cards are indistinguishable from one another.

We define the following encoding scheme to deal with a Boolean value:

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

Given a bit $x \in \{0, 1\}$, when a pair of face-down cards $\boxed{?}\boxed{?}$ describes the value of x with encoding scheme (1), it is called a *commitment* to x and is expressed as

$$\underbrace{\boxed{?}\boxed{?}}_x. \quad (2)$$

For a commitment to $x \in \{0, 1\}$, we sometimes write

$$\underbrace{\boxed{?}}_{x^0} \quad \underbrace{\boxed{?}}_{x^1}$$

instead of expression (2), where $x^0 := x$ and $x^1 := \bar{x}$. In other words, we sometimes use a one-card encoding scheme, $\boxed{\clubsuit} = 0$, $\boxed{\heartsuit} = 1$, for convenience.

Given commitments to players' private inputs, a card-based protocol is supposed to produce a sequence of cards as its output. *Committed* protocols produce their output as a commitment. For example, any committed AND protocol outputs

$$\underbrace{\boxed{?}\boxed{?}}_{a \wedge b}$$

from input commitments to a and b . On the other hand, *non-committed* protocols produce their output in another form.

Hereafter, for a sequence consisting of $d \in \mathbb{N}$ cards, each card of the sequence is sequentially numbered from the left (position 1, position 2, ..., position d), e.g.,

$$\overset{1}{\boxed{?}} \overset{2}{\boxed{\clubsuit}} \overset{3}{\boxed{\heartsuit}} \cdots \overset{d}{\boxed{?}}.$$

1.2 Our Results

As mentioned above, given commitments to Alice’s bit a and Bob’s bit b together with an additional card \clubsuit , Cheung’s AND protocol produces a commitment to $a \wedge b$ with a probability of $1/2$; when it fails, the players have to create their input commitments again. This paper shows that in the last step of Cheung’s AND protocol, a commitment to the OR value $a \vee b$ is also obtained when the protocol succeeds in producing a commitment to $a \wedge b$. Next, we show that, even when the protocol fails, we can still evaluate the AND value (more precisely, any Boolean function) by slightly modifying the last step of the protocol. Thus, the improved protocol never fails to compute the AND value.

Furthermore, we present two five-card copy protocols using unequal division shuffle. Because the most efficient copy protocol currently known requires six cards (MS09), our new protocols improve upon the existing results in terms of the number of required cards, as shown in Table 1. Note that our protocols require an average of two trials. We also design a general copy protocol that produces n copied commitments using unequal division shuffle for an arbitrary $n \geq 3$.

Table 1: Protocols for making two copied commitments

	# of cards	Type of shuffle	Avg. # of trials
(CK94)	8	RC	1
(MS09)	6	RBC	1
Ours (§ 4)	5	UDS	2

RC: Random Cut, RBC: Random Bisection Cut,
UDS: Unequal Division Shuffle

The remainder of this paper is organized as follows. Section 2 first introduces Cheung’s AND protocol along with known shuffle operations and then presents a more general definition of unequal division shuffle. Section 3 describes our slight modification to the last step of Cheung’s AND protocol to expand its functionality. Section 4 proposes two new copy protocols that outperform the previous protocols in terms of the number of required cards. Section 5 presents a general copy protocol. Section 6 demonstrates how to practically implement unequal division shuffle with physical card cases. Finally, Section 7 summarizes our findings and concludes the paper.

An earlier version of this study was presented and appeared as an LNCS paper (NNH⁺15). The present paper is substantially extended as compared to the LNCS paper: this paper extends the previous results to designing a general copy protocol that produces n copied commitments, and also demonstrates how to practically implement unequal division shuffle in details. Sections 5 and 6 are devoted to these new results.

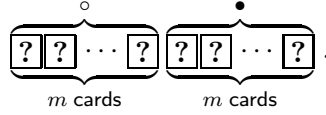
2 Card Shuffling Operations and Known Protocol

In this section, we first introduce a random bisection cut (MS09). Then, we give a general definition of unequal division shuffle. Finally, we introduce Cheung’s AND proto-

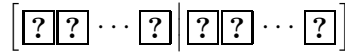
col (CHL13).

2.1 Random Bisection Cut

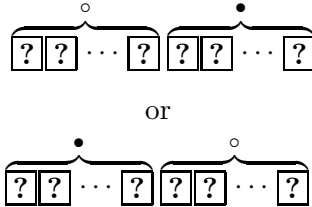
Suppose that there is a sequence of $2m$ face-down cards for some $m \in \mathbf{N}$:



Then, a *random bisection cut* (MS09) (denoted by $[\cdot|\cdot]$)



means that we bisect the sequence and randomly switch the two portions (of size m). Thus, the result of the operation will be either



where each occurs with a probability of exactly $1/2$.

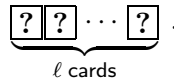
The random bisection cut enables us to significantly reduce the number of required cards and trials for secure computations; the four-card non-committed AND protocol (MKS12), the six-card committed AND protocol (MS09), and the four-card committed XOR protocol (MS09) all employ random bisection cuts. Using random bisection cuts, we can also construct a six-card copy protocol (MS09) (as seen in Table 1), adder protocols (MAS13), protocols for any three-variable symmetric functions (NMS13), and so on.

Whereas the efficient committed AND protocol (MS09) using a random bisection cut requires six cards as stated above, Cheung et al. introduced unequal division shuffle whereby they constructed a five-card committed AND protocol that works with a probability of $1/2$. Its details are presented in the next two subsections.

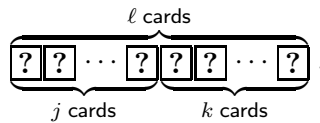
2.2 Unequal Division Shuffle

Here, we present a formal definition of unequal division shuffle, which first appeared in Cheung's AND protocol (CHL13).

Suppose that there is a sequence of $\ell \geq 3$ ($\ell \in \mathbf{N}$) face-down cards:



Divide it into two portions of unequal sizes, say, j cards and k cards, where $j+k = \ell$, $j \neq k$, as follows:



We consider an operation that randomly switches these two portions of unequal sizes; we refer to it as *unequal division shuffle* or (j, k) -*division shuffle* (denoted by $[\cdot|\cdot]$) :

$$\left[\underbrace{[\boxed{?} \boxed{?} \cdots \boxed{?}]}_{j \text{ cards}} \mid \underbrace{[\boxed{?} \boxed{?} \cdots \boxed{?}]}_{k \text{ cards}} \right].$$

Thus, the result of the operation will be either

$$\underbrace{[\boxed{?} \boxed{?} \cdots \boxed{?}]}_{j \text{ cards}} \underbrace{[\boxed{?} \boxed{?} \cdots \boxed{?}]}_{k \text{ cards}}$$


OR

$$\underbrace{[\boxed{?} \boxed{?} \cdots \boxed{?}]}_{k \text{ cards}} \underbrace{[\boxed{?} \boxed{?} \cdots \boxed{?}]}_{j \text{ cards}},$$

where each case occurs with a probability of exactly $1/2$.

We demonstrate feasible implementations (for humans) of unequal division shuffle in Section 6.

2.3 Cheung's AND Protocol

In this subsection, we introduce Cheung's AND protocol. It requires an additional card  to produce a commitment to $a \wedge b$ from two commitments

$$\underbrace{[\boxed{?} \boxed{?}]}_a \underbrace{[\boxed{?} \boxed{?}]}_b$$

placed by Alice and Bob, respectively. As mentioned in Section 2.2, the protocol uses unequal division shuffle, specifically $(2, 3)$ -division shuffle, as follows.


1. Arrange the cards of the two input commitments and the additional card as

$$\underbrace{[\boxed{?}]}_{a^0} \underbrace{[\boxed{?} \clubsuit]}_{a^1} \underbrace{[\boxed{?}]}_{b^0} \underbrace{[\boxed{?}]}_{b^1} \underbrace{[\boxed{?}]}_{b^1}.$$


2. Apply $(2, 3)$ -division shuffle:

$$\left[\boxed{?} \boxed{?} \mid \boxed{?} \boxed{?} \boxed{?} \right].$$

3. Reveal the card at position 1.

- (a) If the card is , then the cards at positions 2 and 3 constitute a commitment to $a \wedge b$:

$$\clubsuit \underbrace{[\boxed{?} \boxed{?} \boxed{?}]}_{a \wedge b}.$$

- (b) If the card is , then Alice and Bob create input commitments again to restart the protocol.

This is Cheung's AND protocol. As seen from step 3, it fails with a probability of $1/2$ (in this case, we have to start from scratch). We verify the correctness of the protocol in the next section.

3 Improved Cheung's AND Protocol

In this section, we discuss Cheung's AND protocol and change its last step to develop an improved protocol.

Here, we confirm the correctness of Cheung's AND protocol. As mentioned in Section 2.3, the input to Cheung's AND protocol consists of commitments to $a, b \in \{0, 1\}$ along with an additional card \clubsuit . There are two possibilities due to the outcome of (2, 3)-division shuffle:

$$\underbrace{\boxed{?}}_{a^0} \underbrace{\boxed{?}}_{\clubsuit} \underbrace{\boxed{?}}_{a^1} \underbrace{\boxed{?}}_{b^0} \underbrace{\boxed{?}}_{b^1} \text{ and } \underbrace{\boxed{?}}_{a^1} \underbrace{\boxed{?}}_{b^0} \underbrace{\boxed{?}}_{b^1} \underbrace{\boxed{?}}_{a^0} \underbrace{\boxed{?}}_{\clubsuit}.$$

We enumerate all possibilities of input and card sequences after step 2 of the protocol in Table 2 (recall encoding scheme (1)). Looking at the cards at positions 2 and 3 when the card at position 1 is \clubsuit in Table 2, we can easily confirm the correctness of the protocol, i.e., the cards at positions 2 and 3 surely constitute a commitment to $a \wedge b$.

Table 2: All possibilities of input and card sequences after step 2

Input (a, b)	Card sequences									
	a^0	\clubsuit	a^1	b^0	b^1	a^1	b^0	b^1	a^0	\clubsuit
(0, 0)	\clubsuit	\clubsuit	\heartsuit	\clubsuit	\heartsuit	\heartsuit	\clubsuit	\heartsuit	\clubsuit	\clubsuit
(0, 1)	\clubsuit	\clubsuit	\heartsuit	\heartsuit	\clubsuit	\heartsuit	\heartsuit	\clubsuit	\clubsuit	\clubsuit
(1, 0)	\heartsuit	\clubsuit	\clubsuit	\clubsuit	\heartsuit	\clubsuit	\clubsuit	\heartsuit	\heartsuit	\clubsuit
(1, 1)	\heartsuit	\clubsuit	\clubsuit	\heartsuit	\clubsuit	\clubsuit	\heartsuit	\clubsuit	\heartsuit	\clubsuit

In the remainder of this section, we analyze Cheung's AND protocol further to obtain an improved protocol.

3.1 Bonus Commitment to OR

When we succeed in obtaining a commitment to $a \wedge b$, i.e., when the card at position 1 is \clubsuit in the last step of Cheung's AND protocol, we are also able to simultaneously obtain a commitment to the OR value $a \vee b$. Thus, as indicated in Table 2, if the card at position 1 is \clubsuit , then the cards at positions 4 and 5 constitute a commitment to $a \vee b$.





3.2 In Case of Failure

Suppose that the card at position 1 is \heartsuit in the last step of Cheung's AND protocol. This means that the AND computation failed and we have to start from scratch, i.e., Alice and Bob need to create their private input commitments again. However, we show that they need not do so: they can evaluate the AND value even when Cheung's AND protocol fails, as follows.



From Table 2, if the card at position 1 is \heartsuit , the sequence of five cards



$$\heartsuit \boxed{?} \boxed{?} \boxed{?} \boxed{?} \tag{3}$$

Table 3: Possible sequences when Cheung's AND protocol fails

Input (a, b)	Sequence of five cards
$(0, 0)$	
$(0, 1)$	
$(1, 0)$	
$(1, 1)$	

is one of the four possibilities shown in Table 3, depending on the value of (a, b) .

Therefore, the card at position 4 indicates the value of $a \wedge b$, i.e., if the card at position 4 is , then $a \wedge b = 0$, and if the card is , then $a \wedge b = 1$. Note that opening the card does not reveal any information about the inputs a and b besides the value of $a \wedge b$. Thus, Cheung's AND protocol does not fail to compute the AND value.

Actually, we can compute any Boolean function $f(a, b)$ in a non-committed format, given the sequence (3) above, as follows. Note that, as seen in Table 3, the position of the face-down card  (which is between 2 and 5) uniquely determines the value of the input (a, b) . We shuffle all cards at positions corresponding to $f(a, b) = 1$ (possibly one card as in the case of $f(a, b) = a \wedge b$) and reveal all these cards. If  appears anywhere, then $f(a, b) = 1$; otherwise, $f(a, b) = 0$. Thus, we can evaluate the desired function (in a non-committed format).

3.3 Improved Protocol

From the discussion above, we have the following improved protocol.


1. Arrange the five cards as follows:

$$\underbrace{\boxed{?}}_{a^0} \underbrace{\boxed{?}}_{\clubsuit} \underbrace{\boxed{?}}_{a^1} \underbrace{\boxed{?}}_{b^0} \underbrace{\boxed{?}}_{b^1}.$$



2. Apply (2,3)-division shuffle:

$$\left[\boxed{?} \boxed{?} \mid \boxed{?} \boxed{?} \boxed{?} \right].$$

3. Reveal the card at position 1.

- (a) If the card is , then the cards at positions 2 and 3 constitute a commitment to $a \wedge b$; moreover, the cards at positions 4 and 5 constitute a commitment to $a \vee b$:

$$\underbrace{\boxed{\clubsuit} \boxed{?} \boxed{?}}_{a \wedge b} \underbrace{\boxed{?} \boxed{?}}_{a \vee b}.$$

- (b) If the card is , then we can evaluate any desired Boolean function $f(a, b)$. Shuffle all cards at positions corresponding to $f(a, b) = 1$ and reveal them. If  appears, then $f(a, b) = 1$; otherwise, $f(a, b) = 0$.

4 Five-Card Copy Protocols

In this section (and the next section), we focus on protocols for copying a commitment.

From Table 1, using the six-card copy protocol (MS09), a commitment to bit $a \in \{0, 1\}$ can be copied with four additional cards:

$$\underbrace{[\?] [\?] [\clubsuit] [\clubsuit] [\heartsuit] [\heartsuit]}_a \rightarrow \underbrace{[\?] [\?] [\?] [\?] [\clubsuit] [\heartsuit]}_a \underbrace{[\?] [\?] [\?] [\?] [\clubsuit] [\heartsuit]}_a.$$

This is the most efficient protocol currently known for copying. In contrast, we prove that three additional cards (two \clubsuit s and one \heartsuit) are sufficient by proposing a five-card copy protocol using unequal division shuffle. We also propose another copy protocol that has fewer steps by considering a different shuffle.

4.1 Copy Protocol Using Unequal Division Shuffle

Given a commitment

$$\underbrace{[\?] [\?] }_a$$

together with additional cards $[\clubsuit] [\clubsuit] [\heartsuit]$, our protocol makes two copied commitments, as follows.

1. Arrange the five cards as

$$\underbrace{[\?] [\?] [\?] [\?] [\?] }_{\begin{matrix} \clubsuit & a^0 & \heartsuit & a^1 & \clubsuit \end{matrix}}.$$

2. Apply (2, 3)-division shuffle:

$$[[\?] [\?] | [\?] [\?] [\?]].$$

3. Rearrange the sequence of five cards as

$$\begin{array}{c} [\?] [\?] [\?] [\?] [\?] \\ \swarrow \quad \searrow \\ [\?] [\?] [\?] [\?] [\?] \end{array}.$$

4. Reveal the card at position 5.

- (a) If the card is \clubsuit , then we have two commitments to a as follows:

$$\underbrace{[\?] [\?] [\?] [\?] }_a \underbrace{[\clubsuit]}_a.$$

- (b) If the card is \heartsuit , then we have

$$\underbrace{[\?] [\?] [\?] [\?] }_{\bar{a}} [\heartsuit].$$

Swap the cards at positions 1 and 2 to obtain a commitment to a . After revealing the cards at positions 3 and 4 (which must be $\clubsuit \clubsuit$), return to step 1.

After step 3, there are two possibilities due to the shuffle outcome: the sequence of five cards is either $\clubsuit \heartsuit \clubsuit a^1 a^0$ or $\heartsuit \clubsuit a^0 \clubsuit a^1$. Table 4 enumerates all possibilities of input and card sequences after step 3 of the protocol. As can be easily seen in the table, we surely have two copied commitments in step 4(a). Note that opening the card at position 5 does not reveal any information about the input a . Thus, we have designed a five-card copy protocol that improves upon the previous results in terms of the number of required cards. It should be noted that the protocol is a Las Vegas algorithm with an average of two trials.

Table 4: Possible sequences after step 3 of our first copy protocol

Input	Card sequences	
a	$\clubsuit \heartsuit \clubsuit a^1 a^0$	$\heartsuit \clubsuit a^0 \clubsuit a^1$
0	$\clubsuit \heartsuit \clubsuit \heartsuit \clubsuit$	$\heartsuit \clubsuit \clubsuit \heartsuit$
1	$\clubsuit \heartsuit \clubsuit \clubsuit \heartsuit$	$\heartsuit \clubsuit \heartsuit \clubsuit \clubsuit$

4.2 Copy Protocol Using Double Unequal Division Shuffle

In this subsection, we reduce the number of steps for achieving copy computation by modifying the unequal division shuffle approach.

Remember that (2,3)-division shuffle changes the order of the two portions:

$$\begin{array}{|c|c|} \hline 1 & 2 \\ \hline \boxed{?} & \boxed{?} \\ \hline \end{array} : \begin{array}{|c|c|c|} \hline 3 & 4 & 5 \\ \hline \boxed{?} & \boxed{?} & \boxed{?} \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|} \hline 3 & 4 & 5 \\ \hline \boxed{?} & \boxed{?} & \boxed{?} \\ \hline \end{array} : \begin{array}{|c|c|} \hline 1 & 2 \\ \hline \boxed{?} & \boxed{?} \\ \hline \end{array} .$$

Here, we consider a further division of the three-card portion:

$$\begin{array}{|c|c|} \hline 3 & 4 \\ \hline \boxed{?} & \boxed{?} \\ \hline \end{array} : \begin{array}{|c|} \hline 5 \\ \hline \boxed{?} \\ \hline \end{array} \begin{array}{|c|c|} \hline 1 & 2 \\ \hline \boxed{?} & \boxed{?} \\ \hline \end{array} \rightarrow \begin{array}{|c|} \hline 5 \\ \hline \boxed{?} \\ \hline \end{array} : \begin{array}{|c|c|} \hline 3 & 4 \\ \hline \boxed{?} & \boxed{?} \\ \hline \end{array} \begin{array}{|c|c|} \hline 1 & 2 \\ \hline \boxed{?} & \boxed{?} \\ \hline \end{array} .$$

Thus, given a sequence of five cards

$$\begin{array}{|c|c|c|c|c|} \hline 1 & 2 & 3 & 4 & 5 \\ \hline \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ \hline \end{array},$$

a shuffle operation resulting in either

$$\begin{array}{|c|c|c|c|c|} \hline 1 & 2 & 3 & 4 & 5 \\ \hline \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ \hline \end{array} \text{ or } \begin{array}{|c|c|c|c|c|} \hline 5 & 3 & 4 & 1 & 2 \\ \hline \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ \hline \end{array}$$

is called *double unequal division shuffle*.

Using such a shuffle, we can avoid rearranging the cards in step 3 of the protocol presented in Section 4.1, as follows.

1. Arrange the five cards as

$$\underbrace{\boxed{?} \boxed{?}}_{a^0} \underbrace{\boxed{?} \boxed{?}}_{\clubsuit} \underbrace{\boxed{?}}_{\heartsuit} \underbrace{\boxed{?} \boxed{?}}_{\clubsuit} \underbrace{\boxed{?}}_{a^1} .$$

2. Apply double unequal division shuffle:

$$\left[\boxed{?} \boxed{?} \mid \boxed{?} \boxed{?} \mid \boxed{?} \right].$$

3. Reveal the card at position 1.

(a) If the card is \clubsuit , then we have two commitments to a :

$$\boxed{\clubsuit} \underbrace{\boxed{?} \boxed{?}}_a \underbrace{\boxed{?} \boxed{?}}_a.$$

(b) If the card is \heartsuit , then we have

$$\boxed{\heartsuit} \underbrace{\boxed{?} \boxed{?} \boxed{?} \boxed{?}}_{\bar{a}}.$$

Swap the cards at positions 2 and 3 to obtain a commitment to a . After revealing the cards at positions 4 and 5, return to step 1.

This protocol has two possibilities after step 2: the sequence of five cards is either $a^0 \clubsuit \heartsuit \clubsuit a^1$ or $a^1 \heartsuit \clubsuit a^0 \clubsuit$. Table 5 confirms the correctness of the protocol.

Table 5: Possible sequences after step 2 of our second copy protocol

Input	Card sequences				
	a	$a^0 \clubsuit \heartsuit \clubsuit a^1$	$a^1 \heartsuit \clubsuit a^0 \clubsuit$		
0	$\clubsuit \heartsuit \heartsuit \heartsuit \heartsuit$	$\heartsuit \heartsuit \heartsuit \heartsuit \heartsuit$			
1	$\heartsuit \heartsuit \heartsuit \heartsuit \heartsuit$	$\heartsuit \heartsuit \heartsuit \heartsuit \heartsuit$			

In the next section, we will extend this protocol to a general protocol that can produce more than two copied commitments.

5 General Copy Protocol

In this section, we propose a general copy protocol, that produces n identical copied commitments from a given commitment to $a \in \{0, 1\}$ using $2n + 1$ cards, where $n \geq 2$.

As comparison to the previous results is shown in Table 6, this protocol reduces the number of cards required to obtain n copied commitments, whereas the average number of trials increases. Note that, n commitments to a can be obtained using $2n + 1$ cards by repeatedly performing the five-card copy protocols presented in Section 4; however, the following protocol requires fewer steps and trials.

Our protocol is a generalization of the five-card copy protocol constructed in Section 4.2. Thus, we employ a type of double unequal division shuffle. Specifically, given a sequence of $2n + 1$ cards

$$\boxed{?}^1 \boxed{?}^2 \boxed{?}^3 \boxed{?}^4 \cdots \boxed{?}^{2n} \boxed{?}^{2n+1},$$

we use the following double unequal division shuffle:

$$\left[\begin{array}{c|c} \overset{1}{?} & \overset{2}{?} \\ \hline \boxed{?} & \boxed{?} \end{array} \mid \begin{array}{c|c} \overset{3}{?} & \overset{4}{?} \\ \hline \boxed{?} & \boxed{?} \end{array} \cdots \begin{array}{c|c} \overset{2n}{?} & \overset{2n+1}{?} \\ \hline \boxed{?} & \boxed{?} \end{array} \right].$$

Therefore, the result of the operation must be either

$$\overset{1}{?} \overset{2}{?} \overset{3}{?} \overset{4}{?} \cdots \overset{2n}{?} \overset{2n+1}{?} \quad \text{or} \quad \overset{2n+1}{?} \overset{3}{?} \overset{4}{?} \cdots \overset{2n}{?} \overset{1}{?} \overset{2}{?},$$

where each occurs with a probability of exactly $1/2$.

The following is the procedure of our general copy protocol.

Table 6: Copy protocols for making n commitments

	# of cards	Type of shuffle	Avg. # of trials
(CK94)	$2n + 4$	RC	1
(MS09)	$2n + 2$	RBC	1
Ours (§ 5)	$2n + 1$	DUDS	2

RC: Random Cut, RBC: Random Bisection Cut, DUDS: Double Unequal Division Shuffle

1. Arrange a given commitment to a and $2n - 1$ additional cards as

$$\begin{array}{c} \overbrace{\boxed{?} \boxed{?} \boxed{?} \boxed{?} \cdots \boxed{?} \boxed{?} \boxed{?}}^{2n-1 \text{ cards}} \\ \underbrace{\boxed{?}}_{a^0} \underbrace{\boxed{?}}_{\clubsuit} \underbrace{\boxed{?}}_{\heartsuit} \underbrace{\boxed{?}}_{\clubsuit} \cdots \underbrace{\boxed{?}}_{\heartsuit} \underbrace{\boxed{?}}_{\clubsuit} \underbrace{\boxed{?}}_{a^1} \end{array}.$$

2. Apply the following double unequal division shuffle:

$$\left[\begin{array}{c|c} \boxed{?} & \boxed{?} \\ \hline \underbrace{\boxed{?}}_{a^0} & \underbrace{\boxed{?}}_{\clubsuit} \end{array} \mid \begin{array}{c|c} \boxed{?} & \boxed{?} \\ \hline \underbrace{\boxed{?}}_{\heartsuit} & \underbrace{\boxed{?}}_{\clubsuit} \end{array} \cdots \begin{array}{c|c} \boxed{?} & \boxed{?} \\ \hline \underbrace{\boxed{?}}_{\heartsuit} & \underbrace{\boxed{?}}_{\clubsuit} \end{array} \mid \begin{array}{c|c} \boxed{?} & \boxed{?} \\ \hline \underbrace{\boxed{?}}_{a^1} & \underbrace{\boxed{?}}_{\clubsuit} \end{array} \right].$$

3. Reveal the card at position 1.

- (a) If the card is \clubsuit , then we have n commitments to a as follows:

$$\begin{array}{c} \overbrace{\clubsuit \boxed{?} \boxed{?} \boxed{?} \boxed{?} \cdots \boxed{?} \boxed{?}}^{2n \text{ cards}} \\ \underbrace{\boxed{?}}_a \underbrace{\boxed{?}}_a \underbrace{\boxed{?}}_a \end{array}.$$

- (b) If the card is \heartsuit , then we have $n - 1$ commitments to negation of a as follows:

$$\begin{array}{c} \overbrace{\heartsuit \boxed{?} \boxed{?} \cdots \boxed{?} \boxed{?} \boxed{?} \boxed{?}}^{2n-2 \text{ cards}} \\ \underbrace{\boxed{?}}_{\bar{a}} \underbrace{\boxed{?}}_{\bar{a}} \end{array}.$$

To obtain one more commitment to a , after revealing the last two cards (which must be $\clubsuit\clubsuit$), execute the five-card copy protocol shown in Section 4.2.

Table 7 shows all possibilities before revealing the card at position 1. Note that this protocol takes an average number of two trials.

Table 7: Possible sequences after step 2 of our general copy protocol

		Card sequences	
a		a^0 ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣	a^1 ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣
0		♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣	♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣
1		♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣	♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣ ♣

6 Implementation of Unequal Division Shuffle and Double Unequal Division Shuffle

This section discusses how to implement unequal division shuffle and double unequal division shuffle with everyday objects.

Note that a random bisection cut (introduced in Section 2.1) can be easily implemented by humans; after bisecting a given card sequence, Alice and Bob take turns to randomly switch the two portions until they are satisfied. On the other hand, if Alice and Bob try to implement unequal division shuffle in the same way, then they will realize the current order of the two portions because of the different sizes of the portions. To avoid such information leakage, we propose to utilize physical cases that satisfy some properties.

Specifically, we consider the card cases shown in Figure 6. Each case can store a portion of cards and has two sliding covers, an upper cover and a lower cover. We assume that the weight of a deck of cards is negligible compared to the case. We think, for instance, that boxes (Figure 6) or envelopes (Figure 6) can be used as such cases.

In the sequel, we implement every unequal division shuffle appearing so far using card cases.

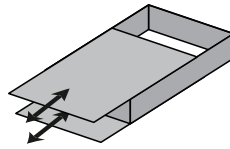


Figure 1: A box suitable for card case

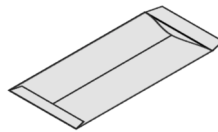


Figure 2: An envelope suitable for card case

6.1 How to Implement the (2,3)-division Shuffle

Here, we propose an implementation of the (2,3)-division shuffle using two cases.

Remember that, after applying the (2,3)-division shuffle

$$\left[\begin{array}{cc|ccc} 1 & 2 & & & \\ \hline ? & ? & ? & ? & ? \end{array} \right],$$

we must have either

$$\begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \\ \hline ? & ? & ? & ? & ? \end{array} \text{ or } \begin{array}{ccccc} 3 & 4 & 5 & 1 & 2 \\ \hline ? & ? & ? & ? & ? \end{array},$$

where each occurs with a probability of $1/2$.

The following steps perform the (2,3)-division shuffle used in the Cheung's AND protocol (Section 2.3), its improved protocol (Section 3.3), and the five-card copy protocol (Section 4.1).

1. Divide a given five-card sequence into a two-card portion and a three-card portion; then, store the first portion in the first case C_1 , and the second portion in the second case C_2 (Figure 3):

$$\begin{array}{cc} 1 & 2 \\ \hline ? & ? \end{array} \rightarrow C_1 \quad \Big| \quad \begin{array}{ccc} 3 & 4 & 5 \\ \hline ? & ? & ? \end{array} \rightarrow C_2.$$

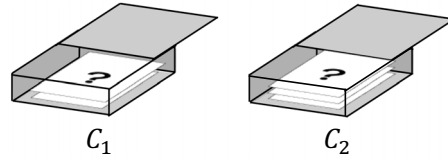


Figure 3: Storing the two portions

2. Switch C_1 and C_2 randomly (Figure 4). This operation results in two possible outcomes:

$$C_1 C_2 \text{ or } C_2 C_1,$$

where each occurs with a probability of $1/2$.

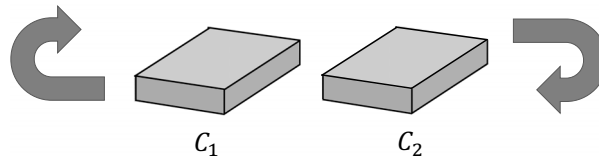


Figure 4: Switching C_1 and C_2 randomly

3. Stack up these cases, as illustrated in Figure 5.
4. Remove the two middle sliding covers simultaneously, as illustrated in Figure 6. Then, we have a sequence of five cards.

As a result of this operation, we have either

$$\begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \\ \hline ? & ? & ? & ? & ? \end{array}$$

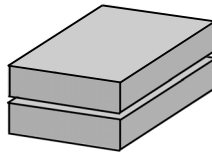


Figure 5: Stacking up the cases

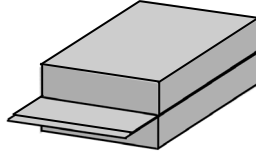


Figure 6: Removing the two covers

(in the case of $C_1 C_2$), or

$$\begin{array}{ccccc} 3 & 4 & 5 & 1 & 2 \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \end{array}$$

(in the case of $C_2 C_1$).

Therefore, the (2,3)-division shuffle can be implemented by humans with card cases.

6.2 Implementation of Double Unequal Division Shuffle Used in Five-Card Copy Protocol

In Section 4.2, a five-card copy protocol using double unequal division shuffle was proposed. We show that it is possible to perform the double unequal division shuffle using three cases.

Remember that the used double unequal division shuffle

$$\left[\begin{array}{cc|cc} \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \end{array} \right] \vdots \boxed{5}$$

results in either

$$\begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \end{array} \quad \text{or} \quad \begin{array}{ccccc} 5 & 3 & 4 & 1 & 2 \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \end{array}.$$

The following steps perform the desired shuffle.

1. Divide a given five-card sequence into two two-card portions and a one-card portion; then, store the first portion in the first case C_1 , the second portion in the second case C_2 , and the third portion in the third case C_3 (Figure 7):

$$\begin{array}{cc} \boxed{1} & \boxed{2} \\ \boxed{?} & \boxed{?} \end{array} \rightarrow C_1 \quad \left| \quad \begin{array}{cc} \boxed{3} & \boxed{4} \\ \boxed{?} & \boxed{?} \end{array} \rightarrow C_2 \quad \left| \quad \boxed{5} \rightarrow C_3.$$

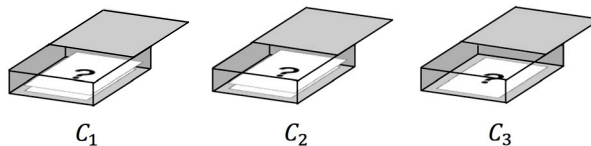


Figure 7: Storing the three portions

2. Switch C_1 and C_3 randomly (Figure 8). This operation results in two possible outcomes:

$$C_1 C_2 C_3 \text{ or } C_3 C_2 C_1,$$

where each occurs with a probability of $1/2$.

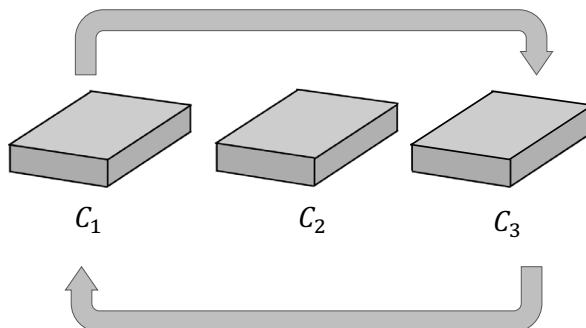


Figure 8: Switching C_1 and C_3 randomly

3. Stack up these cases (without changing the order), as illustrated in Figure 9.

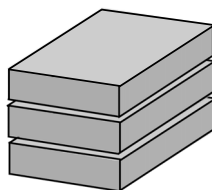


Figure 9: Stacking up the cases

4. Remove all sliding covers except for the top-most and bottom-most covers simultaneously, as illustrated in Figure 10. Then, we have a five-card sequence.

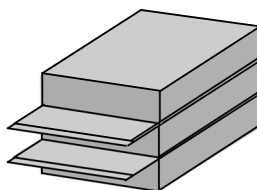


Figure 10: Removing the four covers

As a result of this operation, we have either

1	2	3	4	5
?	?	?	?	?

(in the case of $C_1 C_2 C_3$), or

5	3	4	1	2
?	?	?	?	?

(in the case of $C_3 C_2 C_1$).

Therefore, the double unequal division shuffle can also be implemented.

6.3 Implementation of Double Unequal Division Shuffle Used in General Copy Protocol

We proposed a general copy protocol using $2n + 1$ cards in Section 5. It is also possible to implement the double unequal division shuffle used in the protocol with three cases.

Remember that the used double unequal division shuffle

$$\left[\begin{array}{c|c} \begin{array}{cc} \overset{1}{?} & \overset{2}{?} \\ \boxed{?} & \boxed{?} \end{array} & \begin{array}{ccc} \overset{3}{?} & \overset{4}{?} & \dots \\ \boxed{?} & \boxed{?} & \dots \end{array} \dots \begin{array}{c} \overset{2n}{?} \\ \boxed{?} \end{array} \vdots \begin{array}{c} \overset{2n+1}{?} \\ \boxed{?} \end{array} \end{array} \right]$$

results in either

$$\begin{array}{c} \overset{1}{?} \ \overset{2}{?} \ \overset{3}{?} \ \overset{4}{?} \ \dots \ \overset{2n}{?} \ \overset{2n+1}{?} \\ \boxed{?} \ \boxed{?} \ \boxed{?} \ \boxed{?} \ \dots \ \boxed{?} \ \boxed{?} \end{array} \quad \text{or} \quad \begin{array}{c} \overset{2n+1}{?} \ \overset{3}{?} \ \overset{4}{?} \ \dots \ \overset{2n}{?} \ \overset{1}{?} \ \overset{2}{?} \\ \boxed{?} \ \boxed{?} \ \boxed{?} \ \dots \ \boxed{?} \ \boxed{?} \ \boxed{?} \end{array}.$$

Implementing this shuffle is achieved by almost same operation as the previous subsection. The difference is only the portion to be stored in C_2 . We just substitute $\begin{array}{c} \overset{3}{?} \ \overset{4}{?} \\ \boxed{?} \ \boxed{?} \end{array} \dots \begin{array}{c} \overset{2n}{?} \\ \boxed{?} \end{array}$ for $\begin{array}{c} \overset{3}{?} \ \overset{4}{?} \\ \boxed{?} \ \boxed{?} \end{array}$. Storing the first two cards in C_1 and the last card in C_3 is the same.

Thus, the following steps should be performed. We now use envelopes instead of boxes to illustrate the cases.

1. Divide a given sequence into three portions, and store them in cases C_1 , C_2 , and C_3 , as illustrated in Figure 11:

$$\begin{array}{c|c} \begin{array}{cc} \overset{1}{?} & \overset{2}{?} \\ \boxed{?} & \boxed{?} \end{array} & \begin{array}{ccc} \overset{3}{?} & \overset{4}{?} & \dots \\ \boxed{?} & \boxed{?} & \dots \end{array} \dots \begin{array}{c} \overset{2n}{?} \\ \boxed{?} \end{array} \vdots \begin{array}{c} \overset{2n+1}{?} \\ \boxed{?} \end{array} \end{array}.$$

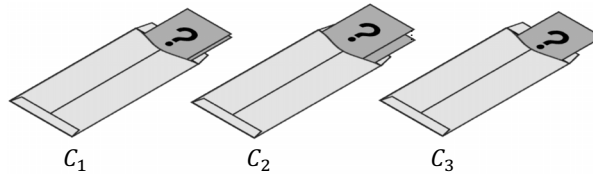


Figure 11: Putting the three portions

2. Switch C_1 and C_3 randomly (Figure 12). This operation results in two possible outcomes:

$$C_1 C_2 C_3 \quad \text{or} \quad C_3 C_2 C_1,$$

where each occurs with a probability of $1/2$.

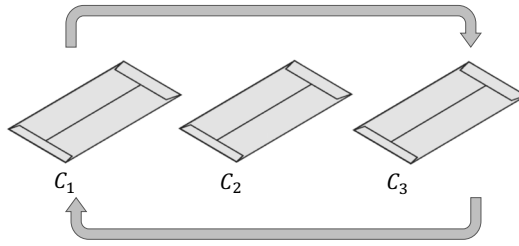


Figure 12: Switching C_1 and C_3 randomly

3. Heap up the three cases (without changing the order), as illustrated in Figure 13.

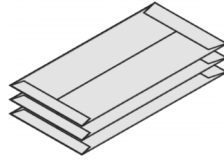


Figure 13: Heaping up the three cases

4. Eject all cards from the envelopes, so as not to change the order of cards and leak any information. We may put the three envelopes in a larger envelope, and eject all the cards inside the larger envelop, as illustrated in Figure 14. Then, we have a sequence of $2n + 1$ cards

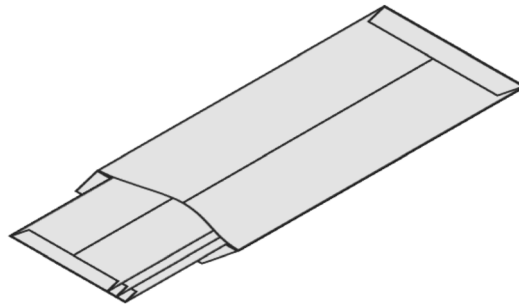
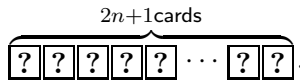


Figure 14: Using a large envelop

One can easily verify the correctness of our implementation.

7 Conclusion

In this paper, we discussed the properties of the AND protocol designed by Cheung et al. and proposed an improved protocol. Although their original protocol produces only

a commitment to the AND value with a probability of $1/2$, our improved protocol either produces commitments to the AND and OR values or evaluates any Boolean function. Thus, the improved protocol does not fail at all.

Furthermore, we proposed two five-card copy protocols that can securely copy an input commitment using three additional cards. Each of our protocols uses unequal division shuffle. Because the most efficient copy protocol currently known requires six cards, our new protocols improve upon the existing results in terms of the number of required cards.

Extending the results, we also designed a general copy protocol that produces n copied commitments using double unequal division shuffle. In addition, we demonstrated how to practically implement unequal division shuffle in details.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers 25289068 and 26330001.

References

- [Boe90] 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, volume 434 of Lecture Notes in Computer Science, pages 208–217. Springer Berlin Heidelberg, 1990.
- [CHL13] Eddie Cheung, Chris Hawthorne, and Patrick Lee. CS 758 project: secure computation with playing cards. http://cslclub.uwaterloo.ca/~cdhawth/static%2Fslashsecure_playing_cards.pdf, 2013.
- [CK94] Claude Crépeau and Joe Kilian. Discreet solitary games. In Douglas R. Stinson, editor, Advances in Cryptology — CRYPTO '93, volume 773 of Lecture Notes in Computer Science, pages 319–330. Springer Berlin Heidelberg, 1994.
- [KWH15] Alexander Koch, Stefan Walzer, and Kevin Härtel. Card-based cryptographic protocols using a minimal number of cards. In Tetsu Iwata and JungHee Cheon, editors, Advances in Cryptology – ASIACRYPT 2015, volume 9452 of Lecture Notes in Computer Science, pages 783–807. Springer Berlin Heidelberg, 2015.
- [MAS13] Takaaki Mizuki, Isaac Kobina Asiedu, and Hideaki Sone. Voting with a logarithmic number of cards. In Giancarlo Mauri, Alberto Dennunzio, Luca Manzoni, and Antonio E. Porreca, editors, Unconventional Computation and Natural Computation, volume 7956 of Lecture Notes in Computer Science, pages 162–173. Springer Berlin Heidelberg, 2013.
- [MKS12] 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, volume 7658 of Lecture Notes in Computer Science, pages 598–606. Springer Berlin Heidelberg, 2012.

- [MS09] Takaaki Mizuki and Hideaki Sone. Six-card secure AND and four-card secure XOR. In Xiaotie Deng, John Edward Hopcroft, and Jinyun Xue, editors, Frontiers in Algorithmics, volume 5598 of Lecture Notes in Computer Science, pages 358–369. Springer Berlin Heidelberg, 2009.
- [NMS13] Takuya Nishida, Takaaki Mizuki, and Hideaki Sone. Securely computing the three-input majority function with eight cards. In Adrian-Horia Dediu, Carlos Martín-Vide, Bianca Truthe, and Miguel A. Vega-Rodríguez, editors, Theory and Practice of Natural Computing, volume 8273 of Lecture Notes in Computer Science, pages 193–204. Springer Berlin Heidelberg, 2013.
- [NNH⁺15] Akihiro Nishimura, Takuya Nishida, Yuichi Hayashi, Takaaki Mizuki, and Hideaki Sone. Five-card secure computations using unequal division shuffle. In Adrian-Horia Dediu, Luis Magdalena, and Carlos Martín-Vide, editors, Theory and Practice of Natural Computing, volume 9477 of Lecture Notes in Computer Science, pages 109–120. Springer International Publishing, 2015.
- [NR98] Valtteri Niemi and Ari Renvall. Secure multiparty computations without computers. Theoretical Computer Science, 191(1–2):173–183, 1998.
- [Sti01] Anton Stiglic. Computations with a deck of cards. Theoretical Computer Science, 259(1–2):671–678, 2001.