Five-Card Secure Computations Using Unequal Division Shuffle*

Akihiro Nishimura[1], Takuya Nishida[1], Yu-ichi Hayashi[2], Takaaki Mizuki[3], and Hideaki Sone[3]

[1] Sone-Mizuki Lab., Graduate School of Information Sciences, Tohoku University
6-3 Aramaki-Aza-Aoba, Aoba, Sendai 980-8578, Japan
[2] Faculty of Engineering, Tohoku Gakuin University
1-13-1 Chuo, Tagajo, Miyagi 985-8537, Japan
[3] Cyberscience Center, Tohoku University
6-3 Aramaki-Aza-Aoba, Aoba, Sendai 980-8578, Japan
tm-paper+card5copw[atmark]g-mail.tohoku-university.jp

Abstract. Card-based cryptographic protocols can perform secure computation of Boolean functions. Cheung et al. recently 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.

Keywords: Cryptography, Card-based protocols, Card games, Cryptography without computers, Real-life hands-on cryptography, Secure multi-party computations

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 (see Table 1), one of which can be selected by them for secure AND computation. For example, if they select the six-card AND protocol [6], they can securely produce

* This paper appears in Proceedings of TPNC 2015. The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-26841-5_9.

a hidden value of $a \wedge b$ using six playing cards, e.g., $\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\heartsuit}\boxed{\heartsuit}\boxed{\heartsuit}$, along with a "random bisection cut."

Recently, Cheung et al. presented an elegant protocol that securely produces a hidden AND value using only five cards ($\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\heartsuit}\boxed{\heartsuit}$); however, it fails with a probability of 1/2 [2] (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 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.

Table 1. Known card-based protocols for secure computation

	# of colors	# of cards	Type of shuffle	Avg. # of trials	Failure rate
∘ *Secure AND in a non-committed format*					
den Boer [1]	2	5	RC	1	0
Mizuki-Kumamoto-Sone [5]	2	4	RBC	1	0
∘ *Secure AND in a committed format*					
Crépeau-Kilian [3]	4	10	RC	6	0
Niemi-Renvall [8]	2	12	RC	2.5	0
Stiglic [10]	2	8	RC	2	0
Mizuki-Sone [6]	2	6	RBC	1	0
Cheung et al. [2] (§2.3)	2	5	UDS	1	1/2
∘ *Secure XOR in a committed format*					
Crépeau-Kilian [3]	4	14	RC	6	0
Mizuki-Uchiike-Sone [7]	2	10	RC	2	0
Mizuki-Sone [6]	2	4	RBC	1	0

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

1.1 Preliminary Notations

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

1. All cards of the same type (black $\boxed{\clubsuit}$ or red $\boxed{\heartsuit}$) 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:

$$\boxed{\clubsuit}\boxed{\heartsuit} = 0,\ \boxed{\heartsuit}\boxed{\clubsuit} = 1. \tag{1}$$

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

$$\underbrace{\boxed{?}\boxed{?}}_{x}. \tag{2}$$

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

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

instead of expression (2), where $x^0 := x$ and $x^1 := \overline{x}$. 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. The *committed* protocols listed in Table 1 produce their output as a commitment. For example, any 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 $n \in \mathbb{N}$ cards, each card of the sequence is sequentially numbered from the left (position 1, position 2, ..., position n), e.g.,

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

1.2 Our Results

As mentioned above, given commitments to Alice's bit a and Bob's bit b together with an additional card $\boxed{\clubsuit}$, 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 [6], our new protocols improve upon the existing results in terms of the number of required cards, as shown in Table 2. Note that our protocols require an average of two trials.

Table 2. Protocols for making two copied commitments

	# of colors	# of cards	Type of shuffle	Avg. # of trials	Failure rate
Crépeau-Kilian [3]	2	8	RC	1	0
Mizuki-Sone [6]	2	6	RBC	1	0
Ours (§4)	2	5	UDS	2	0

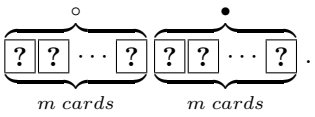
The remainder of this paper is organized as follows. Section 2 first introduces Cheung's AND protocol along with known shuffle operations and then presents a more general definition of unequal division shuffle. Section 3 describes our slight modification to the last step of Cheung's AND protocol to expand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

2 Card Shuffling Operations and Known Protocol

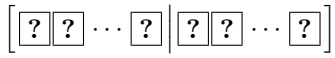
In this section, we first introduce a random bisection cut [6]. Then, we give a general definition of unequal division shuffle. Finally, we introduce Cheung's AND protocol [2].

2.1 Random Bisection Cut

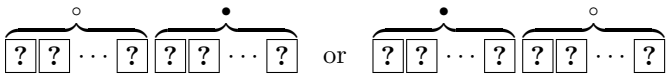
Suppose that there is a sequence of $2m$ face-down cards for some $m \in \mathbb{N}$:



Then, a *random bisection cut* [6] (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. See Table 1 again; the four-card non-committed AND protocol [5], the six-card committed AND protocol [6], and

the four-card committed XOR protocol [6] all employ random bisection cuts. Using random bisection cuts, we can also construct a six-card copy protocol [6] (as seen in Table 2), adder protocols [4], protocols for any three-variable symmetric functions [9], and so on.

Whereas the most efficient committed AND protocol [6] currently known (that always works) uses a random bisection cut and 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 [2].

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

$$\underbrace{?\,?\,\cdots\,?}_{\ell\ cards}\ .$$

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

$$\overbrace{\underbrace{?\,?\,\cdots\,?}_{j\ cards}\ \underbrace{?\,?\,\cdots\,?}_{k\ cards}}^{\ell\ cards}\ .$$

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 $[\cdot|\cdot]$) :

$$\left[\underbrace{?\,?\,\cdots\,?}_{j\ cards}\ \middle|\ \underbrace{?\,?\,\cdots\,?}_{k\ cards}\right].$$

Thus, the result of the operation will be either

$$\underbrace{?\,?\,\cdots\,?}_{j\ cards}\ \underbrace{?\,?\,\cdots\,?}_{k\ cards}\quad \text{or} \quad \underbrace{?\,?\,\cdots\,?}_{k\ cards}\ \underbrace{?\,?\,\cdots\,?}_{j\ cards},$$

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

We demonstrate an implementation of unequal division shuffle in Appendix A.

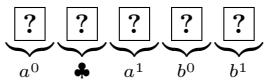
2.3 Cheung's AND Protocol

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

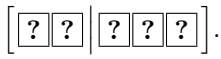
$$\underbrace{?\,?}_{a}\ \underbrace{?\,?}_{b}$$

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.

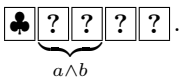
1. Arrange the cards of the two input commitments and the additional card as



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



3. Reveal the card at position 1.
   (a) If the card is ♣, then the cards at positions 2 and 3 constitute a commitment to $a \wedge b$:



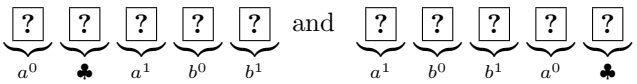
   (b) If the card is ♡, then Alice and Bob create input commitments again to restart the protocol.

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 discussed 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 ♣. There are two possibilities due to the outcome of $(2, 3)$-division shuffle:



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

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

Table 3. All possibilities of input and card sequences after step 2

Input	Card sequences									
(a, b)	a^0 ♣	a^1	b^0	b^1		a^1	b^0	b^1	a^0 ♣	
$(0, 0)$	♣	♣	♡	♣	♡	♡	♣	♡	♣	♣
$(0, 1)$	♣	♣	♡	♡	♣	♡	♡	♣	♣	♣
$(1, 0)$	♡	♣	♣	♣	♡	♣	♣	♡	♡	♣
$(1, 1)$	♡	♣	♣	♡	♣	♣	♡	♣	♡	♣

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 $\boxed{\clubsuit}$ 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 3, if the card at position 1 is $\boxed{\clubsuit}$, 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 $\boxed{\heartsuit}$ 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 3, if the card at position 1 is $\boxed{\heartsuit}$, the sequence of five cards

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

is one of the four possibilities shown in Table 4, depending on the value of (a, b).

Table 4. Possible sequences when Cheung's AND protocol fails

Input (a, b)	Sequence of five cards
$(0, 0)$	$\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\clubsuit}$
$(0, 1)$	$\boxed{\heartsuit}\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\clubsuit}$
$(1, 0)$	$\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\heartsuit}$
$(1, 1)$	$\boxed{\heartsuit}\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\heartsuit}\boxed{\clubsuit}$

Therefore, the card at position 4 indicates the value of $a \wedge b$, i.e., if the card at position 4 is $\boxed{\clubsuit}$, then $a \wedge b = 0$, and if the card is $\boxed{\heartsuit}$, then $a \wedge b = 1$. Note that opening the card does not reveal any information about the inputs a and b

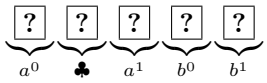
besides the value of $a \wedge b$. Thus, Cheung's AND protocol does not fail to compute the AND value.

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 4, the position of the face-down card $\boxed{\heartsuit}$ (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 $\boxed{\heartsuit}$ appears anywhere, then $f(a, b) = 1$; otherwise, $f(a, b) = 0$. Thus, we can evaluate the desired function (in a non-committed format).

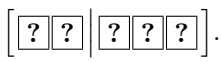
3.3 Improved Protocol

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

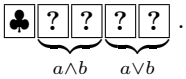
1. Arrange the five cards as follows:



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



3. Reveal the card at position 1.
   (a) If the card is $\boxed{\clubsuit}$, 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 $a \vee b$:

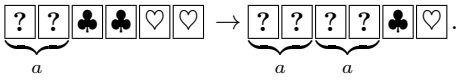


   (b) If the card is $\boxed{\heartsuit}$, then we can evaluate any desired Boolean function $f(a, b)$. Shuffle all cards at positions corresponding to $f(a, b) = 1$ and reveal them. If $\boxed{\heartsuit}$ appears, then $f(a, b) = 1$; otherwise, $f(a, b) = 0$.

4 Five-Card Copy Protocols

In this section, we focus on protocols for copying a commitment.

From Table 2, using the six-card copy protocol [6], a commitment to bit $a \in \{0, 1\}$ can be copied with four additional cards:



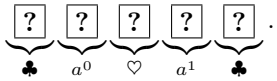
This is the most efficient protocol currently known for copying. In contrast, we prove that three additional cards (two $\boxed{\clubsuit}$s and one $\boxed{\heartsuit}$) are sufficient by proposing a five-card copy protocol using unequal division shuffle. We also propose another copy protocol that has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

4.1 Copy Protocol Using Unequal Division Shuffle

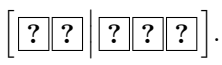
Given a commitment

together with additional cards $\boxed{\clubsuit}\boxed{\clubsuit}\boxed{\heartsuit}$, our protocol makes two copied commitments, as follows.

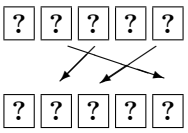
1. Arrange the five cards as



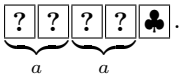
2. Apply $(2,3)$-division shuffle:



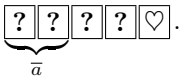
3. Rearrange the sequence of five cards as



4. Reveal the card at position 5.
   (a) If the card is $\boxed{\clubsuit}$, then we have two commitments to a as follows:



   (b) If the card is $\boxed{\heartsuit}$, then we have



   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 $\boxed{\clubsuit}\boxed{\clubsuit}$), 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 5 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 5. Possible sequences after step 3 of our first copy protocol

Input a	Card sequences	
	♣ ♡ ♣ a^1 a^0	♡ ♣ a^0 ♣ a^1
0	♣ ♡ ♣ ♡ ♣	♡ ♣ ♣ ♣ ♡
1	♣ ♡ ♣ ♣ ♡	♡ ♣ ♡ ♣ ♣

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:

$$\overset{1}{?}\,\overset{2}{?} : \overset{3}{?}\,\overset{4}{?}\,\overset{5}{?} \to \overset{3}{?}\,\overset{4}{?}\,\overset{5}{?} : \overset{1}{?}\,\overset{2}{?}.$$

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

$$\overset{3}{?}\,\overset{4}{?} : \overset{5}{?} \quad \overset{1}{?}\,\overset{2}{?} \to \overset{5}{?} : \overset{3}{?}\,\overset{4}{?} \quad \overset{1}{?}\,\overset{2}{?}.$$

Thus, given a sequence of five cards

$$\overset{1}{?}\,\overset{2}{?}\,\overset{3}{?}\,\overset{4}{?}\,\overset{5}{?},$$

a shuffle operation resulting in either

$$\overset{1}{?}\,\overset{2}{?}\,\overset{3}{?}\,\overset{4}{?}\,\overset{5}{?} \quad \text{or} \quad \overset{5}{?}\,\overset{3}{?}\,\overset{4}{?}\,\overset{1}{?}\,\overset{2}{?}$$

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.

1. Arrange the five cards as
$$\underbrace{?}_{a^0}\,\underbrace{?}_{\clubsuit}\,\underbrace{?}_{\heartsuit}\,\underbrace{?}_{\clubsuit}\,\underbrace{?}_{a^1}.$$
2. Apply double unequal division shuffle:
$$\left[?\,?\,|\,?\,:\,?\,?\right].$$
3. Reveal the card at position 1.
   (a) If the card is ♣, then we have two commitments to a:
$$\clubsuit\,\underbrace{?\,?}_{a}\,\underbrace{?\,?}_{a}.$$

(b) If the card is $\boxed{\heartsuit}$, then we have

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

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 6 confirms the correctness of the protocol.

Table 6. Possible sequences after step 2 of our second protocol

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

Although this protocol requires fewer steps, we are not sure whether double unequal division shuffle can be easily implemented by humans.

5 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.

An open problem is whether unequal division shuffle enables us to compute any other function using fewer cards than the existing protocols.

A How to Perform Unequal Division Shuffle

Here, we discuss how to implement unequal division shuffle. We consider the card cases shown in Figure 1. Each case can store a deck 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. To apply unequal division shuffle, we stow each portion in such a case and shuffle these two cases. Then, the cases are stacked one on top of the other. Removing the two middle sliding covers results in the desired sequence.

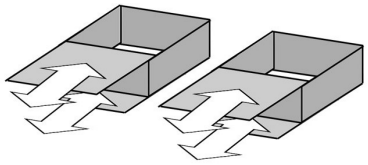


Fig. 1. Card cases suited for unequal division shuffle

Acknowledgments

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

References

1. den Boer, B.: More efficient match-making and satisfiability: the five card trick. In: Quisquater, J.J., Vandewalle, J. (eds.) Advances in Cryptology — EUROCRYPT '89, Lecture Notes in Computer Science, vol. 434, pp. 208–217. Springer Berlin Heidelberg (1990)
2. Cheung, E., Hawthorne, C., Lee, P.: CS 758 project: secure computation with playing cards. http://csclub.uwaterloo.ca/~cdchawth/static/secure_playing_cards.pdf (2013), accessed: 2015-06-22
3. Crépeau, C., Kilian, J.: Discreet solitary games. In: Stinson, D.R. (ed.) Advances in Cryptology — CRYPTO '93, Lecture Notes in Computer Science, vol. 773, pp. 319–330. Springer Berlin Heidelberg (1994)
4. Mizuki, T., Asiedu, I.K., Sone, H.: Voting with a logarithmic number of cards. In: Mauri, G., Dennunzio, A., Manzoni, L., Porreca, A.E. (eds.) Unconventional Computation and Natural Computation, Lecture Notes in Computer Science, vol. 7956, pp. 162–173. Springer Berlin Heidelberg (2013)
5. Mizuki, T., Kumamoto, M., Sone, H.: The five-card trick can be done with four cards. In: Wang, X., Sako, K. (eds.) Advances in Cryptology — ASIACRYPT 2012, Lecture Notes in Computer Science, vol. 7658, pp. 598–606. Springer Berlin Heidelberg (2012)
6. Mizuki, T., Sone, H.: Six-card secure AND and four-card secure XOR. In: Deng, X., Hopcroft, J.E., Xue, J. (eds.) Frontiers in Algorithmics, Lecture Notes in Computer Science, vol. 5598, pp. 358–369. Springer Berlin Heidelberg (2009)
7. Mizuki, T., Uchiike, F., Sone, H.: Securely computing XOR with 10 cards. The Australasian Journal of Combinatorics 36, 279–293 (2006)

8. Niemi, V., Renvall, A.: Secure multiparty computations without computers. Theoretical Computer Science 191(1–2), 173–183 (1998)
9. Nishida, T., Mizuki, T., Sone, H.: Securely computing the three-input majority function with eight cards. In: Dediu, A.H., Martín-Vide, C., Truthe, B., Vega-Rodríguez, M.A. (eds.) Theory and Practice of Natural Computing, Lecture Notes in Computer Science, vol. 8273, pp. 193–204. Springer Berlin Heidelberg (2013)
10. Stiglic, A.: Computations with a deck of cards. Theoretical Computer Science 259(1–2), 671–678 (2001)