

# Universal Construction of Cheater-Identifiable Secret Sharing Against Rushing Cheaters without Honest Majority

Masahito Hayashi<sup>1,2</sup> and Takeshi Koshiba<sup>3</sup>

<sup>1</sup> Graduate School of Mathematics, Nagoya University  
masahito@math.nagoya-u.ac.jp

<sup>2</sup> Centre for Quantum Technologies, National University of Singapore

<sup>3</sup> Faculty of Education and Integrated Arts and Sciences, Waseda University  
tkoshiba@waseda.jp

**Abstract.** For conventional secret sharing, if cheaters can submit possibly forged shares after observing shares of the honest users in the reconstruction phase, they can disturb the protocol and reconstruct the true secret. To overcome the problem, secret sharing scheme with properties of cheater-identification have been proposed. Existing protocols for cheater-identifiable secret sharing assumed non-rushing cheaters or honest majority. In this paper, we remove both conditions simultaneously, and give its universal construction from any secret sharing scheme. To resolve this end, we propose the concepts of “individual identification” and “agreed identification”.

**Keywords:** secret sharing, universal construction, rushing cheater, cheater-identification, without honest majority

## 1 Introduction

Secret sharing is a basic primitive for secure information transmission [1]. It involves a dealer who has a secret  $S$  in the secret set  $\mathcal{S}$  and a set of players. The dealer divides the secret  $S$  into  $n$  shares and distributes the shares to  $n$  players such that if a set of players is qualified then all the players in the set can reconstruct the secret and if the set of players is not qualified then any player in the set cannot obtain any information about the secret. In case of  $(k, n)$ -threshold scheme, any set of  $k$  players can be qualified. Generally, a family  $\mathbb{A}$  of subsets of  $\{1, \dots, n\}$  is the access structure of a secret sharing protocol when any subsets in  $\mathbb{A}$  can reconstruct the secret  $S$  and others can learn nothing about it. It is known that when a family  $\mathbb{A}$  of subset is closed with respect to the union, there exists a secret sharing protocol whose access structure is  $\mathbb{A}$  [2–4]. Further, when a non-qualified set of players obtains a part of information, these schemes are called non-perfect. If a non-perfect scheme has certain threshold properties, then it is called ramp scheme [5–7].

For conventional secret sharing protocols, it is assumed that everyone involved in the protocols is honest or semi-honest. However, in a real scenario, some

**Table 1.** Comparison of proposed CISS protocol with existing CISS protocols

	Number of Cheaters	Rushing	Universal Construction	Efficiency	Flexibility	Large Finite Field
[18]	$t < n/2$	No	No	$O(\ell \log \ell)$	Yes	Needless
[11]	$t < n$	No	Yes	$O(\ell \log \ell)$	No	Needless
[14–16]	$t < k/2$	Yes	No	$O(\ell \log \ell)$	No	Need
[12]	$t < k/2$	Yes	No	$O(\ell \log \ell)$	Yes	Need
Proposed	$t < n$	Yes	Yes	$O(\ell \log \ell)$	Yes	Needless

$n$  is the number of the players.  $t$  is the number of the cheaters.  $k$  is the number of qualified players.  $1 - e^{-\ell}$  is the successful probability to identify the cheaters. Efficiency shows the computational complexity of the protocol. Flexibility is the independence of the choice of the security parameter  $\ell$  from the secret size or the form of original protocol.

participants may maliciously behave in the execution of the protocol. In particular, a part of players may submit incorrect shares so as to yield an incorrect secret in the reconstruction phase. To overcome the problem, additional properties to conventional secret sharing have been considered and new schemes such as cheater-detectable secret sharing (CDSS) [9] and cheater-identifiable secret sharing (CISS) [10] have been proposed. Here, a player that submits incorrect shares is called a cheater. A protocol is called a  $(t, \epsilon)$ -cheater-detectable secret sharing (CDSS) when it detects the existence of cheaters among players involved in reconstruction with probability  $1 - \epsilon$  under the condition that the number of cheaters is not greater than  $t$ . A protocol is called a  $(t, \epsilon)$ -cheater-identifiable secret sharing (CISS) when it identifies who submitted incorrect shares with probability  $1 - \epsilon$  under the condition that the number of cheaters is not greater than  $t$ .

However, cheaters may submit their shares incorrectly *after* observing shares of honest players. Such cheaters is called *rushing* cheaters. The papers [14–16, 12] proposed CISS protocols to properly works against such rushing cheaters. To achieve this task, their sharing phase is composed of two rounds. Unfortunately, these protocols cannot identify the cheaters when the number of cheaters is more than the half of players involved in reconstruction. In this situation, only the protocol in [12] can detect the existence of cheaters without identifying them. Ishai et al [11] proposed another CISS protocol identifying them even when the number of cheaters is more than the half of players involved in reconstruction. To achieve this task, they propose a locally-identifiable secret sharing (LISS), in which a server identifies the cheaters instead of each player, but their LISS is not robust against rushing cheaters. In their protocol, the players submit their shares to the server, and the server recovers the secret and identifies the cheaters for each player. While the server sends each player an information to identify the cheaters, this information depends on the player. That is, this information

is correct only when the player is honest. cheaters. Hence, their identifications do not agree in this protocol.

In a real scenario, it is not easy to prepare the server. Therefore, it is strongly required to propose a protocol to identify the rushing cheaters even when more than half of the players involved in reconstruction are cheaters. In this paper, to resolve this problem, we propose the concepts of “individual identification” and “agreed identification”. A CISS protocol with *individual identification* privately identifies the cheaters so that the identification depends on individual players. A CISS protocol with *agreed identification* commonly identifies the cheaters so that the identification is independent of the player. The difference between these two types of protocols is based on whether their identifications agree or not. The protocol in [11] belongs to the former, and the protocols in [14–16, 12] do to the latter. However, we do not need to distinguish CDSS protocols in this way because there is no advantage even when a CDSS protocol individually detects the existence of the cheaters.

We propose a CISS protocol with individual identification as well as a CISS protocol with agreed identification. Both protocols well work even with rushing cheaters, and the latter is composed of two rounds as well as the protocol in [12]. The former can identify the cheaters even when more than half of the players involved in reconstruction are cheaters. The latter can detect the existence of the cheaters under the same situation, but can identify the cheaters only when less than half of the players in reconstruction are cheaters. When less than half of the players involved in reconstruction are cheaters, even the latter can identify the cheaters. This performance is the same as the protocol given in [12].

Next, we discuss the construction of protocols. Algebraic structures underlie many CISS protocols [14–16, 12] as in the original construction by Shamir. They are limited to  $(k, n)$ -threshold scheme protocols. However, so many efficient secret sharing protocols were proposed when the size of secret is large [6, 4, 5, 8].

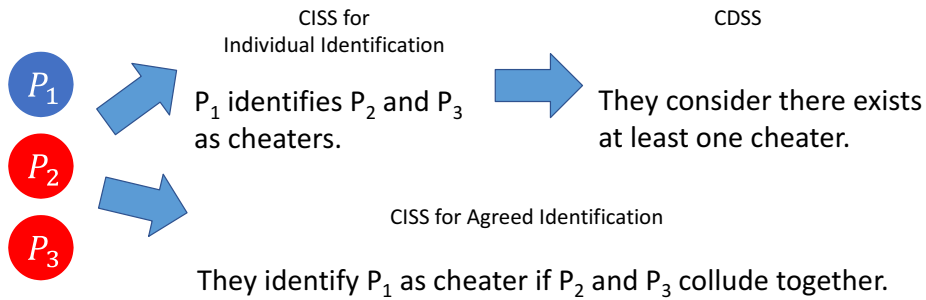
Protocols with general access structure were constructed [2–4]. Also, ramp scheme secret sharing protocols were constructed [5, 6]. Such general secret sharing protocols were not used to in these CISS protocols. Hence, it is desired to construct a CISS protocol by converting an existing secret sharing protocol. Such a construction is called a *universal* construction. The protocol in [11] is universal in this sense. But, it was constructed by converting an existing secret sharing protocol only when the share is given as an element of a finite field. So, to make the scheme more secure, it needs a finite field of larger size. Our construction is universally given when the share of the existing secret sharing protocol is given as an element of vector space of a finite field. That is, it does not require a finite field of large size.

From a practical viewpoint, we need to care about the computational complexity of the protocol. A protocol is efficient when its computational complexity is not so large. When the players identify the cheaters with probability  $1 - e^{-\ell}$ , the computational complexity of the protocols given in [12] is  $O(\ell \log \ell)$ . When the protocol is universally constructed, the total computational complexity depends

on the original secret sharing protocol. In this case, we focus on computational complexity except for the part of the original protocol. In this sense, the protocol in [11] is  $O(\ell \log \ell)$ , and our protocol is also  $O(\ell \log \ell)$ .

However, we cannot necessarily choose the security parameter  $\ell$  freely. In the protocols in [14–16], the security parameter  $\ell$  depends on the size of secret. Hence, it is desired to flexibly choose the security parameter  $\ell$ . We call a protocol flexible, when the security parameter  $\ell$  can be set independently, i.e., independent of the secret size. Flexibility provides the power of partial customization of length of random strings, according to the requirement. The protocol in [12] can flexibly choose the security parameter  $\ell$  by adjusting the finite field with prime size. Also, the protocol in [11] can flexibly choose the security parameter  $\ell$  by adjusting the finite field appearing in the original protocol. Although these protocols offer the flexibility, the security parameter  $\ell$  depends on the size of the finite field. The above computational complexity  $O(\ell \log \ell)$  can be realized by suitable choices of the size of the finite field in these protocols [17]. Hence, the choice of the security parameter  $\ell$  has a certain restriction when we keep the computational complexity  $O(\ell \log \ell)$ . Therefore, it is desired to completely freely choose the security parameter  $\ell$ . Fortunately, our protocol works with any finite field, and the security parameter  $\ell$  can be freely chosen independently of the size of the finite field and the secret size. Therefore, our protocol is flexible and works even with finite field  $\mathbb{F}_2$ , which simplifies the realization. Overall, the comparison of the performances of existing protocols with ours is summarized as Table 1.

The remaining part of this paper is as follows. Section 2 gives our CISS protocol for individual identification. Section 3 shows its security. Section 4 gives our CISS protocols for agreed identification and detection. Section 5 compares the overhead of ours with those of existing protocols.



**Fig. 1.** A Case of majority cheaters. A blue circle expresses a honest player and red circles express cheaters.

## 2 Protocol for Individual Identification

Let  $n$  be the number of players and  $\ell'$  be the security parameter. That is, we will construct our protocol so that the verifier identifies the cheater with probability more than  $1 - q^{-\ell'}$ .

Let  $(\text{Sh}, \text{Rc})$  be a secret sharing protocol realizing access structure  $\mathbb{A}$  with  $\text{Sh} : \mathcal{S} \rightarrow V^n$ , where  $V$  is an  $m$ -dimensional vector space  $\mathbb{F}_q^m$  over a finite field  $\mathbb{F}_q$ . To present our CISS protocol for individual identification based on the protocol  $(\text{Sh}, \text{Rc})$ , we make preparation as follows. For the secret  $S$ , we define the random number  $X_i := \text{Sh}_i(S)$  as the share of the  $j$ -th player, which is sent by the dealer. For  $i \neq j$ , the dealer independently generates  $n(n-1)$  random numbers  $Z_{j,i}$  taking values in  $\mathbb{F}_q^{\ell'}$ . Also, the dealer independently generates  $\ell' \times m$  Toeplitz matrix  $T_j$ . Then, the dealer calculates the random number  $Y_{j,i} := T_j X_i + Z_{j,i}$ . Now, we give our CISS protocol for individual identification as Protocol 1. From Protocol 1, we find that its computational complexity is  $O(\ell' \log \ell')$ .

---

### Protocol 1 CISS protocol for individual identification

---

**STEP 1:** [Dealing] The dealer sends the  $j$ -th player the publishable information  $(X_j, Z_{1,j}, \dots, Z_{j-1,j}, Z_{j+1,j}, \dots, Z_{n,j})$  and the identification-information  $(T_j, Y_{j,1}, \dots, Y_{j,j-1}, Y_{j,j+1}, \dots, Y_{j,n})$ .

**STEP 2:** [Sharing] The players wishing to open the information send their publishable information as follows. The  $i$ -th player sends  $(X_i, Z_{j,i})$  to the  $j$ -th player.

**STEP 3:** [Reconstruction] The  $j$ -th player reconstructs the original information from the collection of  $X'_i$ , where the information received from the  $i$ -th player is  $(X'_i, Z'_{j,i})$ .

**STEP 4:** [Identification] The  $j$ -th player checks whether the relation

$$Y_{j,i} = T_j X'_i + Z'_{j,i} \quad (1)$$

holds.

---

## 3 Security Analysis

As our security analysis, we show the following theorem.

**Theorem 1.** *Protocol 1 is an  $(n-1, q^{-\ell'})$ -CISS protocol realizing access structure  $\mathbb{A}$  with secret space  $\mathcal{S}$  and share space  $\mathcal{S}_i = \mathbb{F}_q^{(2n-1)\ell'+2m-1}$ .*

*Proof.* Since the function  $(X_i, Z_{j,i}) \mapsto T_j X_i + Z_{j,i}$  is a universal<sup>2</sup> hash function with the randomly chosen Toeplitz matrix  $T_j$ , the relation (1) holds with probability smaller than  $q^{-\ell'}$  if the  $j$ -th player makes a cheat. Therefore, even though all of players except for the  $i$ -th player makes cheating even with collusion, the  $i$ -th player can identify who makes cheating with high probability as Fig. 1.

Also, even though several players collude together, they cannot obtain any information for the shares by other players as follows. To see this fact, we assume

that the  $j_1$ -th player, the  $j_2$ -th player,  $\dots$ , the  $j_a$ -th player collude together. We focus on the information  $X_i$  shared by the  $i$ -th player. Since  $Z_{j,i}$  is independent and uniform,  $Y_{j_1,i}, Y_{j_2,i}, \dots, Y_{j_a,i}$  are independent of  $T_{j_1} X_i, T_{j_2} X_i, \dots, T_{j_a} X_i$ . Since they obtain no information for  $T_{j_1} X_i, T_{j_2} X_i, \dots, T_{j_a} X_i$ , they obtain no information  $X_i$ .

Thus, if the original protocol with share  $X_i$  works as secret sharing well, our protocol also works as secret sharing well at least with probability  $1 - q^{-\ell'}$ . Therefore, we obtain the desired statement.

## 4 Protocol for Agreed Identification

Now, we can give our CISS protocol for agreed identification as Protocol 2.

---

### Protocol 2 CISS protocol for agreed identification

---

**STEP 1:** [Dealing] The dealer sends the  $j$ -th player the publishable information  $(X_j, Z_{1,j}, \dots, Z_{j-1,j}, Z_{j+1,j}, \dots, Z_{n,j})$  and the identification-information  $(T_j, Y_{j,1}, \dots, Y_{j,j-1}, Y_{j,j+1}, \dots, Y_{j,n})$ .

**STEP 2:** [Sharing (Round 1)] The players wishing to open the information send their publishable information.

**STEP 3:** [Sharing (Round 2)] The players wishing to open the information send their identification-information.

**STEP 4:** [Reconstruction] The players reconstruct the original information from the collection of  $X'_i$ .

**STEP 5:** [Identification] We employ the majority voting of the results of respective individual identification.

---

Since the majority voting of the results of respective individual verifications identifies who makes cheating if more than half of the players wishing the reconstruction are honest, we have the following theorem.

**Theorem 2.** *Protocol 2 is a  $(\lceil (k-1)/2 \rceil, q^{-\ell'})$ -CISS protocol realizing access structure  $\mathbb{A}$  with secret space  $\mathcal{S}$  and share space  $\mathcal{S}_i = \mathbb{F}_q^{(2n-1)\ell' + 2m-1}$ .*

Modifying Step 5 in Protocol 2 in the following way, we can make a CISS protocol, which is called Protocol 2'. If there exists a player who individually identifies at least one cheater, we consider that there exists a cheater. So, Protocol 2' detects the existence of the cheaters with probability  $1 - q^{-\ell'}$  as Fig. 1, which yields the following theorem.

**Theorem 3.** *Protocol 2' is an  $(n-1, q^{-\ell'})$ -DISS protocol realizing access structure  $\mathbb{A}$  with secret space  $\mathcal{S}$  and share space  $\mathcal{S}_i = \mathbb{F}_q^{(2n-1)\ell' + 2m-1}$ .*

Now, we consider the case when more than half players collude together. We assume that only the  $j_0$ -th player is honest and that the majority cheater, the  $j_1$ -th player,  $\dots$  the  $j_a$ -th player collude together. The cheater, the  $j_v$ -th player

rewrites  $T_{j_v}$ ,  $Z_{j_v, j_w}$  and  $Y_{j_v, j_w}$  for  $1 \leq v \leq a$ ,  $0 \leq w \leq a$  so that  $Y_{j_v, j_w} = T_{j_v} X_{j_w} + Z_{j_v, j_w}$  for  $1 \leq w \leq a$  and  $Y_{j_v, j_0} \neq T_{j_v} X_{j_0} + Z_{j_v, j_0}$ . Due to the majority voting, the agreed identification is that the honest player, the  $j_0$ -th player is a cheater. Therefore, when the majority make cheating, the identification of our CISS protocol for agreed identification is incorrect while the identification of our CISS protocol for individual identification is correct, as Fig. 1.

## 5 Comparison of Overhead

First, we compare the overhead of the protocol in [11] with ours. Let  $u$  be the size of the share of the original secret sharing protocol. When the success probability is  $1 - e^{-\ell}$ , the size of the share of their CISS protocol is greater than  $ue^{-(4n+1)\ell}(n^2(n+1))^{4n+1}$ . That is, their overhead is  $e^{(4n+1)\ell}(n^2(n+1))^{4n+1}$ . However, our protocol has overhead  $e^{(2n-1)\ell}q^{m-1}$ . That is, the their exponential coefficient with respect to the security parameter  $\ell$  is twice as ours.

Next, we compare the overhead of the protocol in [12] with ours. Since their protocol is specified to the  $(k, n)$ -threshold scheme, we translate our overhead to the  $(k, n)$ -threshold scheme. When the secret size is  $|\mathcal{S}|$ , the conventional  $(k, n)$ -threshold scheme has share size  $|\mathcal{S}|p(n)$  for some polynomial  $p$ . When we construct our CISS protocol based on this secret sharing protocol, the share size is  $|\mathcal{S}|p(n)e^{(2n-1)\ell}q^{m-1}$ . That is, its exponential coefficient with respect to the security parameter  $\ell$  is still  $(2n-1)$ . In contrast, the  $(\lceil k/2 \rceil, e^{-\ell})$ -CISS protocol in [12] has share size  $v(n - \lceil k/2 \rceil)^{n+k}e^{(n+k)\ell}$ . That is, its exponential coefficient with respect to the security parameter  $\ell$  is  $n+k$ . So, when  $k$  is close to  $n$ , these two overheads are almost the same.

## 6 Discussion

Firstly, we have proposed to distinguish a CISS protocol for individual identification from a CISS protocol for agreed identification. Then, based on any existing secret sharing protocol, we have universally constructed CISS protocols for individual identification and agreed identification as well as a CDSS protocol. Our CISS protocol for individual identification and our CDSS protocol well work even when more than half of the players involved in reconstruction are cheaters. Our CISS protocol for agreed well works when less than half of the players in reconstruction are cheaters. Our protocols have computational complexity  $O(\ell \log \ell)$  when the probability of successfully identifying (detecting) the cheaters is  $1 - e^{-\ell}$ . We can freely choose the security parameter  $\ell$  independently of the secret size and share size of the original secret sharing protocol. Also, we do not use huge finite fields. That is, we can realize any security parameter  $\ell$  even with the finite field  $\mathbb{F}_2$ . These characteristics simplify the realization. We have checked that the overhead of our protocols are not so huge in comparison with existing protocols.

## Acknowledgments

MH was supported in part by a JSPS Grant-in-Aids for Scientific Research (B) No.16KT0017 and for Scientific Research (A) No.17H01280, the Okawa Research Grant and Kayamori Foundation of Information Science Advancement. TK was supported in part by JSPS Grant-in-Aids for Scientific Research (A) No.16H01705 and for Scientific Research (B) No.17H01695.

## References

1. A. Shamir: How to share a secret, *Communications of the ACM* 22(11):612–613 (1979).
2. M. Itoh, A. Saito, and T. Nishizeki: Multiple assignment scheme for sharing secret, *J. of Cryptology*, vol. 6, pp. 15 – 20, 1993; “Secret sharing schemes realizing general access structure,” *Proceedings of IEEE Global Tele-Communication Conference*, IEEE Press, New Jersey. pp. 993-996, 1987.
3. C. Blundo, A. D. Santis, and U. Vaccaro: On secret sharing schemes, *Information Processing Letters*, no. 65, pp. 25 – 32, 1998.
4. M. Iwamoto and J. Shikata: Secret sharing schemes based on min-entropies, in *Proc. of IEEE International Symposium on Information Theory (ISIT 2014)*, pp.401–405 (2014).
5. G. R. Blakley and Catherine A Meadows: Security of Ramp Schemes, *CRYPTO* 1984: 242-268.
6. H. Yamamoto: On secret sharing systems using  $(k, L, n)$  threshold scheme, *IEICE Trans.*, J68A(9):945–952 (1985), in Japanese. English translation: Electronics and Communications in Japan, Part I, vol. 69, no. 9, pp. 46–54, Scripta Technica, Inc., 1986.
7. Oriol Farràs, Torben Brandt Hansen, Tarik Kaced, and Carles Padró: Optimal Non-perfect Uniform Secret Sharing Schemes, *CRYPTO* (2) 2014: 217-234
8. Hao Chen, Ronald Cramer, Robbert de Haan, and Ignacio Cascudo Pueyo: Strongly Multiplicative Ramp Schemes from High Degree Rational Points on Curves, *EUROCRYPT* 2008.
9. M. Tompa and H. Woll: How to share a secret with cheaters, *J. Cryptology* 1(3):133–138 (1989).
10. R. J. McEliece and D. V. Sarwate: On sharing secrets and Reed-Solomon codes, *Communication of the ACM* 24(9):583–584 (1981).
11. Y. Ishai, R. Ostrovsky, H. Seyalioglu: Identifying cheaters without an honest majority. in *Proc. the 9th Theory of Cryptography Conference (TCC 2012)*, Lecture Notes in Computer Science 7194, pp.21–38, Springer (2012).
12. A. Adhikari, K. Morozov, S. Obana, P. S. Roy, K. Sakurai, and R. Xu: Efficient threshold secret sharing schemes secure against rushing cheaters, in *Proc. the 9th International Conference on Information Theoretic Security (ICITS 2016)*, Lecture Notes in Computer Science 10015, pp.3–23, Springer (2016).
13. K.M. Martin, Challenging the adversary model in secret sharing schemes, In: Coding and Cryptography II, *Proceedings of the Royal Flemish Academy of Belgium for Science and the Arts*, pp. 45-63. 2008.
14. P. S. Roy, A. Adhikari, R. Xu, K. Morozov, and K. Sakurai: An efficient  $t$ -cheater identifiable secret sharing scheme with optimal cheater resiliency, [eprint.iacr.org/2014/628.pdf](https://eprint.iacr.org/2014/628.pdf)



15. R. Xu, K. Morozov, and T. Takagi: On cheater identifiable secret sharing schemes secure against rushing adversary, in *Proc. the 8th International Workshop on Security (IWSEC 2013)*, Lecture Notes in Computer Science 8231, pp.258–271, Springer (2013).
16. R. Xu, K. Morozov, and T. Takagi: Cheater identifiable secret sharing schemes via multi-receiver authentication, in *Proc. the 9th International Workshop on Security (IWSEC 2014)*, Lecture Notes in Computer Science 8639, 72–87, Springer (2014).
17. M. Hayashi and T. Tsurumaru: More efficient privacy amplification with less random seeds via dual universal hash function, *IEEE Transactions on Information Theory*, 62(4):2213–2232 (2016).
18. Tal Rabin Michael Ben-Or: Verifiable Secret Sharing and Multiparty Protocols with Honest Majority, Proceedings of the twenty-first annual ACM symposium on Theory of computing (STOC 1989) pp. 73-85 (1989).