IEICE TRANS. FUNDAMENTALS, VOL.E107–A, NO.3 MARCH 2024

INVITED PAPER *Special Section on Information Theory and Its Applications*

Information-Theoretic Perspectives for Simulation-Based Security in Multi-Party Computation

Mitsugu IWAMOTO[†a)], ***Senior Member***

SUMMARY Information-theoretic security and computational security are fundamental paradigms of security in the theory of cryptography. The two paradigms interact with each other but have shown different progress, which motivates us to explore the intersection between them. In this paper, we focus on *Multi-Party Computation* (MPC) because the security of MPC is formulated by *simulation-based security*, which originates from computational security, even if it requires information-theoretic security. We provide several equivalent formalizations of the security of MPC under a *semi-honest* model from the viewpoints of information theory and statistics. The interpretations of these variants are so natural that they support the other aspects of simulation-based security. Specifically, the variants based on conditional mutual information and sufficient statistics are interesting because security proofs for those variants can be given by information measures and factorization theorem, respectively. To exemplify this, we show several security proofs of BGW (Ben-Or, Goldwasser, Wigderson) protocols, which are basically proved by constructing a simulator.

key words: *information-theoretic security, information measures, sufficient statistics, secure multi-party computation, security formalization, security proofs, BGW protocols*

1. Introduction

1.1 Background

In cryptography, security formalization and security proofs are important because they provide a theoretical basis for ensuring the security of cryptographic primitives and protocols rigorously. Roughly speaking, there are two types of security criteria — information-theoretic security and computational security.

Information-theoretic security was introduced by Shannon [1] in 1948. One of the ground-breaking ideas of [1] is to regard the statistical independence between plaintext and ciphertext as the security against an attacker with *unbounded* computing power. This type of security notion is called *unconditional security* or *information-theoretic security*. Unconditional security is the strongest security among all security criteria. It is a natural and interesting fact that such the strongest security notion could be proposed because the computer was not available at that time.

Computational security was initiated by seminal papers such as Diffie and Hellman [2] and Rivest, Shamir, and Adleman [3]. Computational security is also revolutionary because it assumes that the attacker's ability is limited to a probabilistic polynomial-time Turing machine. This idea offers us many cryptographic functionalities. The first example of such functionality is public-key cryptography and digital signatures [3], and since then, tremendous varieties of cryptographic functionalities have been proposed.

1.2 Motivation

From the history shown above, both security criteria depend on different theoretical foundations; information-theoretic security is based on information theory, whereas computational security is based on computational complexity theory.

Since information-theoretic security is based on information theory, information measures such as Shannon entropy and mutual information can be used to measure the key length, amount of leakage, etc. For example, the mutual information between plaintext and the corresponding ciphertext implies the amount of leakage from the ciphertext because the mutual information is a measure of the statistical dependence of two random variables. While it has the advantage of being able to measure information in terms of quantity, information-theoretic cryptography can only model the behavior of an attacker probabilistically. Hence, for instance, it is generally not easy to measure the leakage of secret information against malicious adversaries.

On the other hand, computational security has its basis in computational complexity theory because the computational complexity of a specific computationally hard problem is used to guarantee the security of cryptosystems. In addition, computational security can utilize the techniques of computational complexity theory, and it succeeds in modeling the behavior of attackers. For example, *semantic security* is the first example of such security formalization [4]. In semantic security, an attacker simulates the one-bit guess of the plaintext using ciphertext. If the advantage of guessing the plaintext using the ciphertext without the plaintext is negligible, then the cryptosystem is considered secure. This type of formalization is called *simulation-based* security, which is very useful in defining the security of public-key cryptosystems. For details of simulation-based security, the reference is [5] is comprehensive.

Attempts are being made to introduce semantic security into information-theoretic security. For instance, *entropic security* [6] is one such attempt, showing that the information-theoretic security could be realized if plaintext is limited by min-entropy and semantic security is employed.

Manuscript received May 1, 2023.
Manuscript revised August 3, 2023.
Manuscript publicized December 1, 2023.

[†]The author is with the University of Electro-Communications, Chofu-shi, 182-8585 Japan.
a) E-mail: mitsugu@uec.ac.jp

DOI: 10.1587/transfun.2023TAI0001

It was pointed out that [7], in symmetric-key cryptography and authentication, there is a case where the formalization of information-theoretic security is strictly stronger than that of the computational one, even if the attacker's computing power is unlimited.

Motivated by the intersection of information-theoretic security and the computational one discussed above, we revisit the security of secure Multi-Party Computation (MPC) because it is formalized by simulation-based security even when information-theoretic security should be guaranteed.

An MPC is a cryptographic protocol that computes a function without disclosing inputs. Since the output is the result of computation, leakage of input information from the output is inevitable. For example, consider the case where three students take a 100-point test, and the average score is 80. The output (average score) suggests that no student scores less than 40 ($= 80 \times 3 - 100 \times 2$). Thus, MPC wants to ensure that nothing about the input is disclosed except for information leaked from the output. In this sense, MPC requires more delicate treatment in its definition of security than simple confidentiality, such as one-time pad [8], [9] and secret-sharing [10], [11]. Simulation-based security is one of the fundamental techniques to formalize such delicate security. Furthermore, simulation-based security is also useful to capture the behavior of malicious adversaries.

Unfortunately, however, as is mentioned in [5, Abstract], simulation-based security is not easy for beginners due to such a sensitive requirement. Therefore, discussing simulation-based security from multiple perspectives, such as information theory and statistics, should be meaningful to find the implications of what we observe from the discussion. For instance, we want to know the interpretation of simulation-based security from information theory to convince that 0 bit leaks in an MPC protocol. As far as the author knows, such discussions on simulation-based security have never been made. Revisiting simulation-based security from multiple viewpoints would deepen our understanding of simulation-based security and yield interesting results for security proofs.

1.3 Contributions

This paper revisits simulation-based security for MPC under a semi-honest model and discusses it from information theory and statistics. The application of our approach to malicious adversaries is important, but it is outside the scope of this paper and is one of the most important future works of this study.

We will present four formulations that are equivalent to simulation-based security. These formulations are based on conditional probabilities, Markov chains, conditional mutual information, and sufficient statistics. While some of these were previously known, we will explicitly highlight their equivalence.

As far as the author knows, the security of MPC has been proved in most cases by a simulation, i.e., constructing a simulator, even under a semi-honest model. For instance, the security of *BGW (Ben-Or, Goldwasser, and Widgerson) protocol* [12], which is one of the fundamental protocols of MPC, is proved by a simulation. Actually, we can see such proofs based on a simulator in [13], [14]. On the other hand, in this paper, we will provide security proofs by two approaches: One is based on information measures, and the other uses the existence of sufficient statistics.

Since the BGW protocols are based on secret-sharing schemes [11], the security proof of BGW in this approach is quite similar to the security proof of secret-sharing schemes [15]. As a result, the proofs by information measures enable us to see that 0 bit leaks out except for the inputs and outputs in BGW. Fisher's factorization theorem proves the existence of sufficient statistics [16], which could be linked to the field of statistics, such as estimation theory. In both types of security proofs, a simulator is not explicitly constructed, which is an intriguing observation.

1.4 Organization

The rest of this paper is organized as follows. Section 2 introduces several notations and definitions of information measures with their properties used in this paper. For readers' convenience and to clarify our claims, the security criteria of secret sharing, Shamir's secret-sharing scheme [11], and its security proof are explained in Sect. 3. We also review the BGW protocols in Sect. 4.

Following the above preparations, we discuss the simulation-based security for MPC in Sect. 5 under a semi-honest model. We first provide the original simulation-based security in Sect. 5.2, followed by its variants in Sect. 5.3. Based on the discussion in Sect. 5, we explain alternative security proofs for BGW protocols. One is based on information measures, and the other is based on sufficient statistics, which will be provided in Sects. 6 and 7, respectively. Section 8 summarizes the conclusion and future work.

2. Preliminaries

2.1 Notations and Definitions

Let $\mathbb{N}$ be the set of natural numbers. For an integer $n \in \mathbb{N}$, define $[n] \coloneqq \{1, 2, \ldots, n\}$. For a vector $x \coloneqq (x_1, x_2, \ldots, x_n)$ and a set $\mathcal{A} \coloneqq \{i_1, i_2, \ldots, i_t\} \subseteq [n]$ $(i_1 < i_2 < \cdots < i_t)$, the vector induced by $\mathcal{A}$ is defined as $x_{\mathcal{A}} \coloneqq (x_{i_1}, x_{i_2}, \ldots, x_{i_t})$. Analogously, given an $n \times m$ matrix $[x_{i,j}]_{1 \le i \le n, 1 \le j \le m}$,

$$x_{\mathcal{A},\mathcal{B}} \coloneqq \begin{bmatrix} x_{i_1,j_1} & x_{i_1,j_2} & \cdots & x_{i_1,j_u} \\ x_{i_2,j_1} & x_{i_2,j_2} & \cdots & x_{i_2,j_u} \\ \vdots & \vdots & \ddots & \vdots \\ x_{i_t,j_1} & x_{i_t,j_2} & \cdots & x_{i_t,j_u} \end{bmatrix},$$

for two sets $\mathcal{A} \coloneqq \{i_1, i_2, \ldots, i_t\} \subseteq [n]$ $(i_1 < i_2 < \cdots < i_t)$ and $\mathcal{B} \coloneqq \{j_1, j_2, \ldots, j_u\} \subseteq [m]$ $(j_1 < j_2 < \cdots < j_u)$.

Throughout the paper, random variables and their instances are represented by uppercase and lowercase letters,

respectively. For a random variable A, the probability distribution associated with A is given by $P_A(\cdot)$. Calligraphic fonts are used to denote sets. For instance, a probability of a random variable A taking a value a over a set $\mathcal{A}$ is given by $P_A(a)$. The complement set of a set $\mathcal{A}$ is denoted by $\overline{\mathcal{A}}$, and the cardinality of $\mathcal{A}$ is denoted by $|\mathcal{A}|$. A finite field is denoted by $\mathbb{F}$.

Shannon entropy [1] of $P_A(\cdot)$ is defined as

$$H(A) := -\sum_{a\in\mathcal{A}} P_A(a)\log P_A(a). \tag{1}$$

Throughout this paper, the base of logarithms is 2.

For two random variables A and B, joint entropy with respect to $P_{A,B}(\cdot,\cdot)$ is defined in a similar manner with (1), and the conditional entropy of A given B is defined by $H(A|B) := \sum_{b\in\mathcal{B}} P_B(b)H(A|B=b)$, where $H(A|B=b)$ is the Shannon entropy associated with the conditional probability of A given $B=b$, i.e., $P_{A|B}(\cdot \mid b)$.

Mutual information between random variables A and B is defined by

$$I(A\wedge B) := \sum_{\substack{a\in\mathcal{A}\\ b\in\mathcal{B}}} P_{A,B}(a,b)\log\frac{P_{A,B}(a,b)}{P_A(a)P_B(b)}.$$

It is well-known that $I(A\wedge B)$ is symmetric with respect to A and B because of the relation

$$I(A\wedge B) = H(A)+H(B)-H(A,B). \tag{2}$$

2.2 Properties of Information Measures

For readers' convenience, we summarize several fundamental results of information measures used in this paper without proof. See [17] for the proofs if necessary.

Proposition 2.1. Let A and B be random variables taking values over the sets $\mathcal{A}$ and $\mathcal{B}$, respectively. Then, the following properties hold.

1. *Cardinality bound:* Shannon entropy is upper-bounded by the logarithm of the cardinality of the domain.

   $$H(A)\le\log|\mathcal{A}|.$$

   The equality holds if and only if A is uniformly distributed over $\mathcal{A}$.
2. *Subadditivity:* It generally holds that

   $$H(A,B)\le H(A)+H(B).$$

   Equality holds if and only if A and B are statistically independent. Note that $H(A,B)=H(A)+H(B)$ is equivalent to $H(B\mid A)=H(B)$ (or $H(A\mid B)=H(A)$).

The following proposition implies the properties of information measures by an information processing represented by a function f.

Proposition 2.2. Let $f:\mathcal{A}\to\mathcal{B}$ be a (deterministic) map. Assume that the probability distribution $P_B(\cdot)$ is induced from $P_A(\cdot)$ and f by $b=f(a)$, $a\in\mathcal{A}$, $b\in\mathcal{B}$. Then, the following properties hold:

1. $H(A,B)=H(A)$, i.e., $H(B\mid A)=0$. Conversely, $H(A,B)=H(A)$ suggests the existence of a map $f:\mathcal{A}\to\mathcal{B}$ such that $B=f(A)$ with probability 1.
2. If f is surjective, it holds that $H(A,X)=H(B,X)$ for arbitrary random variable X (possibly correlated with A and B).

Remark 2.3. Regarding Prop. 2.2–2., $H(A,X)=H(B,X)$ for arbitrary X implies that f is surjective if the domain of f is restricted to the support of P_A. This relation suggests that we can replace A in $H(A,X)$ with B if such a surjection exists. We write about this relationship as

$$A \overset{in}{\longleftrightarrow} B, \tag{3}$$

which will be useful in the later discussion.

3. Shamir's Secret-Sharing Scheme

This paper will discuss the security proofs of MPCs based on secret sharing. Hence, we give a detailed review of secret-sharing schemes in this section for readers' convenience.

3.1 Protocol of Shamir's Secret-Sharing Scheme

Overview: A secret-sharing scheme was independently proposed by Blakley [10] and Shamir [11]. A secret-sharing scheme consists of a dealer and n parties. In the protocol, a dealer encodes a secret into several pieces called *shares*, sent to parties via secure and authenticated channels. The secret can be recovered from a specified set of shares called *qualified set*. On the other hand, we call a share set a set of shares that *forbidden* if it is not allowed to recover the secret. The pair of families of qualified and forbidden sets is called an *access structure* [18]–[20].

In this paper, we consider a simple case called (k,n)-*threshold secret-sharing scheme* (or simply (k,n)-*threshold scheme*), where all share sets with cardinality more than or equal to k are qualified. In contrast, the secret cannot be recovered from the share sets with cardinality less than k in the sense of information-theoretic security. Computational secret-sharing is outside the scope of this paper, but see [21] if the readers are interested.

Syntax: Let n be the number of parties participating in a secret-sharing scheme, and let s be a secret that takes a value in a set $\mathcal{S}$. Denote by v_i $(i=1,2,\ldots,n)$ a share held by i-th party, where v_i takes a value in a set $\mathcal{V}_i$.

Definition 3.1. A secret-sharing scheme consists of the following two algorithms:

- $\mathsf{ShareGen}:\mathcal{S}\times\mathcal{R}\to\mathcal{V}_1\times\cdots\times\mathcal{V}_n$, where $\mathcal{R}$ is the set of random numbers used in the algorithm.
- $\mathsf{Recov}_{\mathcal{A}}:\mathcal{V}_{\mathcal{A}}\to\mathcal{S}$, where $\mathcal{A}\subseteq[n]$ satisfies $|\mathcal{A}|\ge k$.

The (k,n)-threshold secret-sharing schemes must satisfy the following requirements:

Definition 3.2. Let S be a random variable corresponding to the secret, and let $V_1, V_2, \ldots, V_n$ be a set of n shares generated by $\mathsf{ShareGen}$ and S. $(\mathsf{ShareGen}, \mathsf{Recov})$ forms a (k,n)-threshold secret-sharing scheme, or simply called a (k,n)-threshold scheme, if the following conditions are met:

1. The secret is recovered correctly, which can be defined as follows (See Prop. 2.2–1.):
$$\forall \mathcal{A} \subseteq [n],\ H(S \mid V_{\mathcal{A}}) = 0,\ \text{ if } |\mathcal{A}| \geq k.$$
2. No information of the secret can be obtained from less than k shares, which can be defined as follows (See Prop. 2.1–2.):
$$\forall C \subseteq [n],\ H(S \mid V_C) = H(S),\ \text{ if } |C| \leq k-1.$$

Shamir's secret-sharing Scheme: It is well-known that polynomial interpolation is available for secret-sharing schemes, which was proposed by Shamir [11]. Share generation $\mathsf{ShareGen}(s; r_1, \ldots, r_{k-1})$ is realized by the following random polynomial with degree $k-1$ with s as a constant term.

$$f_s^{k-1}(x) \coloneqq s + r_1 x^1 + \cdots + r_{k-1} x^{k-1}. \tag{4}$$

Then, $\mathsf{ShareGen}$ is represented as

$$\begin{aligned}&\mathsf{ShareGen}(s; r_1, \ldots, r_{k-1})\\ &\quad = (f_s^{k-1}(1), f_s^{k-1}(2), \ldots, f_s^{k-1}(n))\\ &\quad =: (v_1, v_2, \ldots, v_n)\end{aligned}$$

The recovery algorithm for a set of parties $\mathcal{A} \subseteq [n]$, $|\mathcal{A}| = k$[†], with the tuple of shares $v_{\mathcal{A}}$ is obtained by using Lagrange's interpolation. That is, $f_s^{k-1}(\cdot)$ in (4) is recovered by the following formula.

$$f_s^{k-1}(x) = \sum_{\alpha \in \mathcal{A}} v_\alpha \prod_{\beta \in \mathcal{A}\setminus\{\alpha\}} \frac{x-\beta}{\alpha-\beta}.$$

Since $s = f_s^{k-1}(0)$, the secret s can be computed as

$$\begin{aligned}s = f_s^{k-1}(0) &= \sum_{\alpha \in \mathcal{A}} v_\alpha \prod_{\beta \in \mathcal{A}\setminus\{\alpha\}} \frac{-\beta}{\alpha-\beta}\\ &=: \sum_{u \in \mathcal{A}} v_\alpha \rho_\alpha,\end{aligned} \tag{5}$$

where $\rho_\alpha \coloneqq -\prod_{\beta \in \mathcal{A}\setminus\{\alpha\}} \beta/(\alpha-\beta)$. For a set of indices $\mathcal{A} = \{i_1, i_2, \ldots, i_k\}$, let $\rho_{\mathcal{A}} \coloneqq (v_{i_1}, v_{i_2}, \ldots, v_{i_k})$. Then, (5) is written as

$$s = \langle v_{\mathcal{A}}, \rho_{\mathcal{A}} \rangle, \tag{6}$$

where $\langle \cdot, \cdot \rangle$ stands for the inner product. Hence, the recovery function is given by $\mathsf{Recov}_{\mathcal{A}}(\cdot) \coloneqq \langle \cdot, \rho_{\mathcal{A}} \rangle$. Equation (6) is useful in the BGW protocol for multiplication, as we will see in Sect. 4.2. It is worth noting that the recovery vector $\rho_{\mathcal{A}}$ only depends on $\mathcal{A}$, and hence, we can compute $\rho_{\mathcal{A}}$ before receiving $v_{\mathcal{A}}$.

In closing this section, we introduce one more useful property of linear secret sharing. This property holds most of the linear secret-sharing schemes.

Proposition 3.3. In Shamir's secret-sharing scheme for a secret s, let v_C be a tuple of shares for the set $C \subseteq [n]$ of parties with cardinality $k-1$. Then, for any $i \notin C$, v_i can be computed from (v_C, s). Specifically, there exists a map such that

$$\mu_{C,i}^{k-1} : (v_C, s) \mapsto v_i,\ \text{ for all } i \in [n]. \tag{7}$$

Proof. Let $\mathcal{A} = C \cup \{i\}$. Then, given (v_C, s), we can solve a linear equation (6) with respect to v_i, and it is obvious that the solution v_i is determined uniquely. □

3.2 Security Proof of Secret Sharing

It is well known that Shamir's secret-sharing schemes guarantee information-theoretic security. The following is a security proof of Shamir's secret-sharing based on information measures.

Proposition 3.4 ([11], [15])**.** Shamir's secret-sharing scheme achieves information-theoretic security in the sense that it satisfies Definition 3.2–2.

Proof. For arbitrary set of parties $C = \{i_1, i_2, \ldots, i_{k-1}\} \subseteq [n]$, it holds that

$$\begin{bmatrix} s \\ v_{i_1} \\ v_{i_2} \\ \vdots \\ v_{i_{k-1}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & i_1^1 & i_1^2 & \cdots & i_1^{k-1} \\ 1 & i_2 & i_2^2 & \cdots & i_2^{k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & i_{k-1} & i_{k-1}^2 & \cdots & i_{k-1}^{k-1} \end{bmatrix} \begin{bmatrix} s \\ r_1 \\ r_2 \\ \vdots \\ r_{k-1} \end{bmatrix}. \tag{8}$$

It is easy to see that the matrix in (8) is regular. Hence, there exists a surjection associated with C such that

$$\psi_C : (s, v_{i_1}, v_{i_2}, \ldots, v_{i_{k-1}}) \mapsto (s, r_1, r_2, \ldots, r_{k-1}).$$

Using the notation given by (3), we can write

$$(S, V_C) \overset{in}{\longleftrightarrow} (S, R_{[k-1]}). \tag{9}$$

Then, we have

$$\begin{aligned}&I(V_C \wedge S)\\ &\quad = H(S) + H(V_C) - H(SV_C)\\ &\quad = H(S) + H(V_C) - H(SR_{[k-1]})\\ &\quad = H(S) + H(V_C) - (H(S) + H(R_{[k-1]})\\ &\quad = H(V_C) - H(R_{[k-1]}))\\ &\quad \leq 0,\end{aligned} \tag{10}$$

[†]If $|\mathcal{A}| \geq k+1$, we select arbitrary k shares from $\mathcal{A}$.

where $R_{[k-1]} := (R_1, R_2, \ldots, R_{k-1})$. The first and the third equalities are due to (2), and the second equality holds because of Prop. 2.2–2. by considering (9). The inequality is valid since V_C distributes over $\mathbb{F}^{k-1}$ in conjunction with the cardinality bound in Prop. 2.1–1.

Since $I(V_C \wedge S) \geq 0$ generally holds, we can conclude that $I(V_C \wedge S) = 0$, i.e., share set V_C and the secret S are statistically independent. □

We can see the following proposition as a byproduct of the above proof. This is useful in proving information-theoretic security, including simulation-based security.

Proposition 3.5. In Shamir's secret-sharing schemes, the tuple of shares V_C is uniformly distributed over $\mathbb{F}^{|C|}$ if $|C| \leq k-1$.

Proof. The result $I(V_C \wedge S) = 0$ in Prop. 3.4 implies that the inequality in (10) is actually the equality. Noticing that $v_C, r_{[k-1]} \in \mathbb{F}^{k-1}$, it turns out that the set of shares V_C is uniformly distributed over $\mathbb{F}^{k-1}$. □

4. BGW Protocols

Secure Multi-Party Computation (MPC) was initiated by several seminal papers, Rivest, Shamir, and Adleman [22], Yao [23], and Goldwasser, Micali, and Wigderson [24]. These results proposed MPC protocols computing specific functions, such as the comparison of two numbers.

MPCs computing general functions were initiated later by Goldreich, Micali, and Wigderson [24] and Ben-Or, Goldwasser, and Wigderson [12] under computational and information-theoretic settings, respectively. The protocol proposed by [24] consists of garbled circuits [25] and oblivious transfer [26] for computing Boolean arithmetic circuits, and that by [12] was based on secret-sharing schemes for arithmetic computations. These protocols are secure under the semi-honest model, and we do not consider the malicious setting. See [27] for readers interested in MPC in general, including malicious settings.

Since this paper is concerned with information-theoretic security of MPCs under a semi-honest model, we review hereafter the classical result called *BGW protocols* [12] (see also [28], [29]). We subsequently use Shamir's $(t+1, n)$–threshold secret-sharing schemes underlying the BGW protocol because we want to construct the MPC secure against collusion of at most t parties. The purpose of the protocols is to transform the shares of inputs into those of outputs without disclosing the inputs.

Specifically, we assume the following scenarios. The first and the second party, i.e., party 1 and party 2, input two inputs $s^{(1)} \in \mathbb{F}$ and $s^{(2)} \in \mathbb{F}$, respectively. Then, they generate the tuples of shares associated with $s^{(1)}$ and $s^{(2)}$. Finally, from these shares, all n parties obtain the outputs $s^{(1)} + s^{(2)}$ and $s^{(1)}s^{(2)}$ in addition protocol π^{add} and multiplication protocol π^{mult}, respectively.

Before going into details, we introduce a useful notation of MPC. For a secret-sharing scheme with a secret $s \in \mathbb{F}$, we denote the set of all shares of the $(t+1, n)$–threshold scheme by

$$[s; r_1, r_2, \ldots, r_t] =: (v_1, v_2, \ldots, v_n)$$

where $v_i = f_s^t(i)$. If the random numbers are clear from the context, we omit them and use an abbreviation $[s]$ to represent $[s; r_1, r_2, \ldots, r_t]$.

4.1 π^{add}: Addition Protocol

We review the addition protocol π^{add} and its properties, where we want to compute the addition of two inputs $s^{(1)}$ and $s^{(2)}$ over $\mathbb{F}$. The protocol consists of two phases: the share generation and distribution phase and the recovery phase.

4.1.1 Share Generation and Distribution Phase

Suppose that the first and the second party input the secrets $s^{(1)} \in \mathbb{F}$ and $s^{(2)} \in \mathbb{F}$, respectively. Then, they generate the following sets of n shares using different random polynomials $f_{s^{(i)}}^t$ of degree t for $i = 1, 2$.

$$(v_1^{(i)}, v_2^{(i)}, \ldots, v_n^{(i)}) := [s^{(i)}; r_{[t]}^{(i)}], \tag{11}$$

where

$$r_{[t]}^{(i)} := (r_1^{(i)}, r_2^{(i)}, \ldots, r_t^{(i)}).$$

Then, party $i \in \{1, 2\}$ sends $v_j^{(i)}$ to the j-th party via secure and authenticated channels.

From (9) in Prop. 2.1–2., we note that

$$(S^{(i)}, V_C^{(i)}) \overset{in}{\longleftrightarrow} (S^{(i)}, R_C^{(i)}), \quad \text{for } i = 1, 2. \tag{12}$$

4.1.2 Recovery Phase

Party $i \in [n]$, generates the new share by

$$w_i := v_i^{(1)} + v_i^{(2)}, \quad i = 1, 2, \ldots, n. \tag{13}$$

Note that this addition of shares can be performed *locally*, i.e., without additional communication.

From the linearity, it is easy to see that $w_{[n]}$ is a tuple of shares with the secret $s^{(1)} + s^{(2)}$. Concretely, it holds that

$$[s^{(1)} + s^{(2)}; r_{[t]}^{(1)} + r_{[t]}^{(2)}] = (w_1, w_2, \ldots, w_n),$$

which is a tuple of shares of $(t+1, n)$–threshold scheme with the secret $s^{(1)} + s^{(2)}$. Recalling (6), for any set of parties $\mathcal{A} \subseteq [n]$ with $|\mathcal{A}| = t + 1$, the output $s := s^{(1)} + s^{(2)}$ can be computed as

$$s = \langle w_{\mathcal{A}}, \rho_{\mathcal{A}} \rangle.$$

Note that, in this protocol, each input $s^{(1)}$ and $s^{(2)}$ is not revealed before generating the output $s = s^{(1)} + s^{(2)}$.

Recalling (7) in Prop. 3.3, we observe that there exists a map

$$\mu^t_{C,i} : (w_C, s) \mapsto w_i, \quad \text{for all } i \in [n], \tag{14}$$

for an arbitrary set C with $|C| = t$ and any $i \notin C$.

4.2 Multiplication Protocol

We review the multiplication protocol π^{mult} and its properties, where we want to compute the multiplication of two inputs $s^{(1)}$ and $s^{(2)}$ over $\mathbb{F}$. The protocol consists of three phases: the share generation and distribution phase, the threshold reduction phase, and the recovery phase.

As described later, the protocol π^{mult} works only when $n \geq 2t + 1$. Hence, for simplicity, we assume that $n = 2t + 1$ although π^{mult} works in the case where $n > 2t + 1$ analogously.

4.2.1 Share Generation and Distribution Phase

The strategy of computing the multiplication of two secrets is the same as the addition protocol. Concretely, generate

$$w_i \coloneqq v_i^{(1)} v_i^{(2)}, \quad \text{for } i = 1, 2, \ldots, n, \tag{15}$$

where $(v_i^{(1)})_{i=1}^n$ and $(v_i^{(2)})_{i=1}^n$ are the tuples of $(t+1, n)$–threshold schemes defined by (11).

Then, $(w_1, w_2, \ldots, w_n)$ is the tuple of shares for the secret $s \coloneqq s^{(1)}s^{(2)}$ because it the constant term of $v_i^{(1)} v_i^{(2)} = f^t_{s^{(1)}}(x) f^t_{s^{(2)}}(x)$. However, the threshold of the shares is no longer $t + 1$ but is $\tau + 1 \coloneqq 2t + 1 (= n)$ due to the multiplication of random polynomials of degree t.

Recalling (6) in the discussion of Sect. 3, the secret s can be recovered by:

$$s = \langle \rho_{[n]}, w_{[n]} \rangle. \tag{16}$$

4.2.2 Threshold Reduction Phase

Our goal is to generate the shares with threshold $t + 1$ with secret $s = s^{(1)}s^{(2)}$ without revealing $s^{(1)}$ nor $s^{(2)}$. Hence, the reduction of the threshold from $\tau + 1$ to $t + 1$ is necessary, which is realized by re-sharing the shares $w_{[n]}$. This is the central idea of the BGW protocol for multiplication.

The re-sharing technique is described as follows. We share each w_i using $(t+1, n)$–threshold scheme and the shares are denoted by

$$(w_{i,1}, w_{i,2}, \ldots, w_{i,n}) \coloneqq [w_i; r_{i,[t]}], \quad \text{for } i \in [n], \tag{17}$$

where we define

$$r_{i,[t]} \coloneqq (r_{i,1}, r_{i,2}, \ldots, r_{i,t}).$$

Note that $r_{i,[t]}$ is a tuple of random numbers uniformly generated by player i from the set $\mathbb{F}^t$.

Now, recalling the discussion in Sect. 3 again, it holds from (6) that

$$w_i = \langle w_{i,\mathcal{A}}, \rho_{\mathcal{A}} \rangle, \quad \text{for all } i \in [n], \tag{18}$$

for arbitrary set $\mathcal{A} \subseteq [n]$ such that $|\mathcal{A}| = t + 1$. Furthermore, from (9), we also have

$$(W_i, R_{i,[t]}) \xleftrightarrow{in} (W_i, W_{i,C}), \quad \text{for all } i \in [n], \tag{19}$$

for arbitrary set $C \subseteq [n]$ such that $|C| = t$.

The value of $w_{i,j}$ is transmitted from party i to party j using secure and authenticated channels. As a result, party j holds the corresponding shares:

$$w_{[n],j} \coloneqq (w_{1,j}, w_{2,j}, \ldots, w_{n,j}). \tag{20}$$

Hence, the set $C = \{j_1, j_2, \ldots, j_t\} \subseteq [n]$ receives the shares $w_{[n]\setminus C, C}$.

4.2.3 Recovery Phase

The key observation in BGW's degree reduction is the following relation, obtained by combining (16) and (18). For $\mathcal{A} = \{j_1, j_2, \ldots, j_{t+1}\}$, substituting (18) into (16), we have

$$\begin{aligned} s &= s^{(1)}s^{(2)} \\ &\overset{\text{(a)}}{=} \langle \rho_{[n]}, w_{[n]} \rangle \\ &= \langle \rho_{[n]}, (w_1, w_2, \ldots, w_n) \rangle \\ &\overset{\text{(b)}}{=} \rho_{[n]} \begin{bmatrix} w_{1,j_1} & w_{1,j_2} & \cdots & w_{1,j_{t+1}} \\ w_{2,j_1} & w_{2,j_2} & \cdots & w_{2,j_{t+1}} \\ \vdots & & & \\ w_{n,j_1} & w_{n,j_2} & \cdots & w_{n,j_{t+1}} \end{bmatrix} \rho_{\mathcal{A}}^\top \\ &= \rho_{[n]} \left(w_{[n],\mathcal{A}} \right) \rho_{\mathcal{A}}^\top, \end{aligned} \tag{21}$$

where the marked equalities (a) and (b) follow from (16) and (18), respectively.

Here, observe the v-th column of the $n \times (t + 1)$ matrix $w_{[n],\mathcal{A}}$ in (21) is (20) for $j = j_v$, which is the tuple of shares held by the party j_v. Therefore, the computation

$$v_{j_v} \coloneqq \langle \rho_{[n]}, w_{[n],j_v} \rangle, \tag{22}$$

can be executed *locally* by each party $j_v \in \mathcal{A}$. Furthermore, it turns out from (21) for $v_{\mathcal{A}} = (v_{j_1}, v_{j_2}, \ldots, v_{j_{t+1}})$ that

$$s = \langle \rho_{\mathcal{A}}, v_{\mathcal{A}} \rangle,$$

hold, which implies that $v_{\mathcal{A}}$ computed by (22) is the tuple of $t + 1$ shares of $(t + 1, n)$–threshold scheme with secret $s = s^{(1)}s^{(2)}$. Therefore, in a similar manner with (14), there exists a map

$$\mu^t_{C,i} : (v_C, s) \mapsto v_i, \tag{23}$$

for an arbitrary set C with $|C| = t$ and any $i \notin C$.

5. Simulation-Based Security for MPC and Its Variants

So far, we described BGW protocols for addition and multiplication, which rely on Shamir's secret-sharing schemes.

We have established that both of these protocols are correct. This section discusses the simulation-based security of MPC in a semi-honest model under information-theoretic security in order to provide alternative security proofs for BGW protocols in the later sections. We first introduce a *view* for defining the security.

5.1 Views

In formalizing the security of MPC, *views* play a crucial role. Views for a set of players $C \subseteq [n]$ are a set of random variables that consist of inputs, random numbers C generates, and the transcripts C receives from $\overline{C}$. Specifically, the view of the i-th player is defined as

$$\Phi_i := (X_i, R_i; T_i^{(1)}, T_i^{(2)}, \ldots, T_i^{(\ell)}),$$

where X_i, R_i, and $T_i^{(u)}$ are random variables corresponding to the i-th party's input, random numbers for the i-th party, and a transcript that the i-th party receives at the u-th transmission ($1 \le u \le \ell$), respectively. For example, in the case of addition and multiplication protocols explained in Sects. 4.1 and 4.2, the views for the set of parties C such that $|C| = t$ are

$$\Phi_C^{\mathrm{add}} := \left(X_C; V_C^{(1)}, V_C^{(2)}, W_i\right),$$
$$\Phi_C^{\mathrm{mult}} := \left(X_C, R_{C,[t]}; V_C^{(1)}, V_C^{(2)}, W_{[n]\setminus C,C}, V_i\right),$$

respectively, where the party i is selected outside the set C for recovering the secret s in a recovery phase.

Remark 5.1. Generally, a view consists of random variables, sometimes depending on the situation. We clarify whether the views are actual values or random variables by lower and upper cases, respectively. In particular, the random variable corresponding to the view Φ_i given the input x_i is denoted by $\Phi_i(x_i)$. Note that the probability distribution of $\Phi_i(x_i)$ follows the conditional probability distribution $P_{\Phi_i|X_i}(\cdot|x_i)$.

5.2 Simulation-Based Security

In defining the security of MPC, we have to guarantee that no information *beyond the inputs and outputs* leaks to a set of corrupted parties. To formulate this notion, so-called *simulation-based security* is useful. The definition of simulation-based security follows [5], [13], [14].

Let x_i and y_i be the input and output of the i-th player. We also define $\omega_i := (x_i, y_i)$. Simulation-based security states that there exists a probabilistic algorithm $\mathsf{Sim}(\cdot)$ that simulates the tuple of random variables Φ_C by using the C's inputs and outputs view $\omega_C := (x_C, y_C)$. Note that the views actually depend on *all* inputs and outputs $\omega_{[n]} = (x_{[n]}, y_{[n]})$.

Definition 5.2 ([5], [13], [14])**.** An MPC protocol π is called *t-private* if a set of $C \subseteq [n]$ of corrupted parties satisfies the following: If $|C| \le t$, there exists a probabilistic algorithm $\mathsf{Sim}(\cdot)$ that simulates the view with respect to $w_{[n]}$. Concretely, it holds that

$$\forall w_{[n]}, \mathsf{Sim}(\omega_C) \equiv \Phi_C(\omega_{[n]}), \tag{24}$$

where for two random variables A and B, $A \equiv B$ means that random variables A and B are equivalent, i.e., A and B are perfectly indistinguishable.

The computing power of the simulator $\mathsf{Sim}(\cdot)$ will be discussed in the next section (Remark 5.3).

5.3 Variants of Simulation-Based Security

The simulation-based security introduced in the previous section is traditional, and we can see the same definitions in [5], [13], [14], for instance. In this section, we show several variants of simulation-based security from the information theory perspective.

5.3.1 Formalization via Conditional Probability

The conditional probability distribution associated with the random variable $\Phi_C(\omega_{[n]})$ in the left hand side of (24) is $P_{\Phi_C|\Omega_{[n]}}(\cdot \mid \omega_{[n]})$. Hence, the existence of the simulator $\mathsf{Sim}(\cdot)$ is equivalent to the existence of conditional probability distribution $P_{\Phi_C|\Omega_C}(\cdot \mid \omega_C)$ such that

$$\forall \varphi_C, \forall \omega_{[n]}, P_{\Phi_C|\Omega_{[n]}}(\varphi_C|\omega_{[n]}) = P_{\Phi_C|\Omega_C}(\varphi_C|\omega_C). \tag{25}$$

Note that $P_{\Phi_C|\Omega_{[n]}}(\cdot \mid \omega_{[n]})$ can be determined from the protocol. Therefore, if the designed protocol π satisfies (25), the simulator $\mathsf{Sim}(\cdot)$ that satisfies (24) does exist.

Remark 5.3 (Comparable Security)**.** The computing power of the corrupted parties is essential to define security. Usually, a simulator should be a probabilistic polynomial-time algorithm for simulating random variables. However, in the case of MPC, the computing power of the corrupted parties depends on that of the simulator.

Formally, this is called *comparable security* [30]. Since we are now considering the information-theoretic MPC, the computing power of corrupted parties is also unlimited. Hence, the computational power of the simulator $\mathsf{Sim}(\cdot)$ is unlimited, which validates that a conditional probability can characterize the simulator.

On the other hand, if the computing power of $\mathsf{Sim}(\cdot)$ is limited, then the equivalence between (24) and (25) does not hold in general. Therefore, exploring the condition where (24) and (25) are equivalent for the computationally-bounded simulator is interesting.

5.3.2 Formalization via Markov Chain

Considering the fact that $(\omega_C, \omega_{\overline{C}})$ is identical with $\omega_{[n]}$, (25) implies that $\Omega_{\overline{C}}$ and Φ_C are conditionally independent given Ω_C. In other words, $\Omega_{\overline{C}}$, Ω_C, and Φ_C form a Markov chain in this order, which is specifically written as

$$\Omega_{\overline{C}} \multimap \Omega_C \multimap \Phi_C. \tag{26}$$

5.3.3 Formalization via Conditional Mutual Information

It is well known that the Markov chain given by (26) has the equivalent form such that

$$I(\Omega_{\overline{C}} \wedge \Phi_C \mid \Omega_C) = 0. \tag{27}$$

For instance, see [17] for details.

The implication of (27) is that the corrupted parties' view Φ_C contains no information about the honest parties' inputs and outputs $\Omega_{\overline{C}}$ except C's inputs and outputs Ω_C, which seems intuitively understandable.

Note that the representation of information measures was presented, for instance, in [31] for defining t-private computation[†], which is essentially the same as (27), although the relation to the simulation-based security is not mentioned.

5.3.4 Formalization via Sufficient Statistics

The Markov chain given by (26) also offers an important statistical observation upon MPC in terms of *sufficient statistics*. The definition of sufficient statistics is as follows [16], [17]:

Definition 5.4 (Sufficient statistics [16], [17])**.** Suppose we have a family of parametrized probability distributions $\{P_{X_\theta}\}_\theta$. Then, for a random variable X, a statistic $\sigma(X)$ is *sufficient* for θ if it contains all the information in X about θ.

The intuitive meaning of sufficient statistics is that $\sigma(X)$ is *sufficient* for guessing the parameter θ instead of using X since $\sigma(X)$ contains all the information in X about θ.

There are several equivalent definitions of sufficient statistics. This paper introduces information-theoretic characterization of sufficient statistics as follows [17].

Definition 5.5 ([17])**.** For a random variable X, a statistic $\sigma(X)$ is *sufficient* for θ if it holds for arbitrary distribution of θ that

$$\theta \multimap \sigma(X) \multimap X. \tag{28}$$

Note that the order of data processing is θ, X, and $\sigma(X)$, i.e., it holds that, for arbitrary distribution of θ,

$$\theta \multimap X \multimap \sigma(X). \tag{29}$$

Hence, sufficient statistic $\sigma(X)$ satisfies (28) and (29) simultaneously.

Returning to the protocols of MPC, observe that Φ_C can be computed from Ω_C without using $\Omega_{\overline{C}}$. Hence, we have a Markov chain such that

$$\Omega_{\overline{C}} \multimap \Phi_C \multimap \Omega_C. \tag{30}$$

[†]A t-private computation is a kind of MPC, where n party computes a function with n inputs and an output secure against at most t collusion of the parties. BGW protocols are also t-private computation.

Combining (30) with (26), we can conclude that Ω_C is a sufficient statistic with respect to $\omega_{\overline{C}}$ by regarding $\Omega_C = \sigma(\Phi_C)$ as a statistics of Φ_C.

The intuitive meaning of this observation is that the inputs and outputs Ω_C for the corrupted party C is *sufficient* for guessing the honest parties inputs and outputs $\omega_{\overline{C}}$ instead of using C's view Φ_C, i.e., Φ_C is useless for guessing $\omega_{\overline{C}}$.

5.4 Toward the Security Proofs on BGW Protocols

In the following sections, we will prove the security of the protocols π^{add} and π^{mult} using the variants of simulation-based security presented in the previous section. For this purpose, we summarize the previous Sects. 5.2 and 5.3 as the following theorem.

We note that we construct the simulator in order to prove (i), which is the standard manner in proving the security of MPCs. These simulators are presented, for instance, in [13], [14]; hence, we omit such proof. Rather, we are interested in (iv) and (v), i.e., the security proofs based on information measures and sufficient statistics since they are outside the scope of the standard discussion of simulation-based security.

Theorem 5.6. For all $C \subseteq [n]$ corrupted parties with $|C| = t$, the following (i)–(v) are equivalent.

(i) Simulation-based Security defined by (24) in Definition 5.2.
(ii) $\forall \varphi_C, \forall \omega_{[n]}, P_{\Phi_C|\Omega_{[n]}}(\varphi_C \mid \omega_{[n]}) = P_{\Phi_C|\Omega_C}(\varphi_C \mid \omega_C)$ as defined in (25).
(iii) $\Omega_{\overline{C}} \multimap \Omega_C \multimap \Phi_C$ as defined in (26).
(iv) $I(\Omega_{\overline{C}} \wedge \Phi_C | \Omega_C) = 0$ as defined in (27).
(v) Ω_C is a sufficient statistic with respect to $\omega_{\overline{C}}$ by regarding $\Omega_C = \sigma(\Phi_C)$ as a statistics of Φ_C.

Without loss of generality, we assume in the following proofs that the first and the second players input $s^{(1)}$ and $s^{(2)}$, respectively. We also assume that $C \in \{3, 4, \ldots, n\}$ and $i \notin C$.

6. Security Proofs of MPC by Information Measures

This section is devoted to proving the t-privacy on the addition and multiplication protocols of BGW in the sense of Theorem 5.6–(iv), i.e., from the view of information measures.

Theorem 6.1. The addition protocol π^{add} and the multiplication protocol π^{mult} are t-private in the sense of (iv) in Theorem 5.6.

Hereafter, we prove Theorem 6.1 on π^{add} and π^{mult}. Through these proofs, we can see that the security of BGW protocols can be shown by exploiting the same techniques to prove the security of secret-sharing schemes.

6.1 Security Proofs on π^{add}

Proof of Theorem 6.1 on π^{add}. To prove t-privacy of π^{add},

we will show for a set of parties C with cardinality t that

$$I(\Omega_{\overline{C}} \wedge \Phi_C^{\mathrm{add}} \mid \Omega_C) = 0, \tag{31}$$

where, for $S := S^{(1)} + S^{(2)}$ and $i \notin C$, we set[†]

$$\begin{aligned}
\Omega_C &:= (\perp, S), \\
\Omega_{\overline{C}} &:= \left((S^{(1)}, S^{(2)}), S\right), \\
\Phi_C^{\mathrm{add}} &:= \left(\perp; V_C^{(1)}, V_C^{(2)}, W_i\right).
\end{aligned}$$

To prove (31), we will show that the conditional mutual information is not positive. For this purpose, we decompose the conditional mutual information such that

$$\begin{aligned}
&I(\Omega_{\overline{C}} \wedge \Phi_C^{\mathrm{add}} \mid \Omega_C) \\
&= I\left(((S^{(1)}, S^{(2)}), S) \wedge (V_C^{(1)}, V_C^{(2)}, W_i) \mid S\right) \\
&= H\left(V_C^{(1)}, V_C^{(2)}, W_i \mid S\right) \\
&\quad - H\left(V_C^{(1)}, V_C^{(2)}, W_i \mid S^{(1)}, S^{(2)}, S\right).
\end{aligned} \tag{32}$$

The first term of (32) is upper-bounded as

$$\begin{aligned}
&H\left(V_C^{(1)}, V_C^{(2)}, W_i, S\right) \\
&\overset{(a)}{=} H\left(V_C^{(1)}, V_C^{(2)}, W_C, W_i, S\right) \\
&\overset{(b)}{=} H\left(V_C^{(1)}, V_C^{(2)}, W_C, S\right) \\
&\overset{(c)}{=} H\left(V_C^{(1)}, V_C^{(2)}, S\right) \\
&\le H(V_C^{(1)}) + H(V_C^{(2)}) + H(S),
\end{aligned} \tag{33}$$

where the last inequality holds due to the subadditivity of the entropy function (Prop. 2.1–2.), and the marked equalities are due to the following facts in conjunction with Prop. 2.2–1.:

(a) From (13), we can compute W_C from $(V_C^{(1)}, V_C^{(2)})$. Hence, W_C can be included without changing Shannon entropy.
(b) From (14), we can compute W_i from (W_C, S). Hence, we can exclude W_i without changing Shannon entropy.
(c) The same reason as (a).

Recalling that $t = |C|$, it is easy to see that the following cardinality bounds (Prop. 2.1–1.) hold because $V_C^{(1)}$ and $V_C^{(2)}$ distribute over $\mathbb{F}^t$.

$$\begin{aligned}
H(V_C^{(1)}) &\le t \log |\mathbb{F}|, \\
H(V_C^{(2)}) &\le t \log |\mathbb{F}|.
\end{aligned}$$

Hence, (33) is upper-bounded as

$$H\left(V_C^{(1)}, V_C^{(2)}, W_i, S\right) \le 2t \log |\mathbb{F}| + H(S),$$

which yields

$$H\left(V_C^{(1)}, V_C^{(2)}, W_i \mid S\right) \le 2t \log |\mathbb{F}|. \tag{34}$$

Next, we evaluate the second term of (32).

$$\begin{aligned}
&H\left(V_C^{(1)}, V_C^{(2)}, W_i, S^{(1)}, S^{(2)}, S\right) \\
&\overset{(d)}{=} H\left(V_C^{(1)}, V_C^{(2)}, S^{(1)}, S^{(2)}, S\right) \\
&\overset{(e)}{=} H\left(R_{[t]}^{(1)}, R_{[t]}^{(2)}, S^{(1)}, S^{(2)}, S\right) \\
&\overset{(f)}{=} H(R_{[t]}^{(1)}) + H(R_{[t]}^{(2)}) + H(S, S^{(1)}, S^{(2)}) \\
&= 2t \log |\mathbb{F}| + H(S, S^{(1)}, S^{(2)}),
\end{aligned}$$

where the marked equalities hold because of the following reasons:

(d) The same reason with (a) and (b) in (33).
(e) From (12), we can replace $V_C^{(i)}$ with $R_{[t]}^{(i)}$ for $i = 1, 2$.
(f) The random variables $R_{[t]}^{(1)}$, $R_{[t]}^{(2)}$, and $(S, S^{(1)}, S^{(2)})$ are statistically independent.

Therefore, the second term of (32) is calculated as:

$$H\left(V_C^{(1)}, V_C^{(2)}, W_i \mid S^{(1)}, S^{(2)}, S\right) = 2t \log |\mathbb{F}| \tag{35}$$

Combining (34) and (35), we obtain

$$I(\Omega_{\overline{C}} \wedge \Phi_C^{\mathrm{add}} \mid \Omega_C) \le 0.$$

From the non-negativity of the conditional mutual information, we obtain (31). □

6.2 Security Proofs on π^{mult}

Proof of Theorem 6.1 on π^{mult}. In the case of multiplication protocol, recall that $n = \tau + 1$.

To prove that the t-privacy of π^{mult}, we will show that

$$I(\Omega_{\overline{C}} \wedge \Phi_C^{\mathrm{mult}} \mid \Omega_C) = 0, \tag{36}$$

where, for $S := S^{(1)}S^{(2)}$ and $i \notin C$, we set

$$\begin{aligned}
\Omega_C &:= (\perp, S), \\
\Omega_{\overline{C}} &:= \left((S^{(1)}, S^{(2)}), S\right), \\
\Phi_C^{\mathrm{mult}} &:= \left(\perp, R_{C,[t]}; V_C^{(1)}, V_C^{(2)}, W_{[n]\setminus C,C}, V_i\right).
\end{aligned}$$

To prove (36), we will show that the conditional mutual information is not positive, which is a similar strategy in proving Prop. 3.4.

First, we decompose the conditional mutual information as follows:

$$\begin{aligned}
&I(\Omega_{\overline{C}} \wedge \Phi_C^{\mathrm{mult}} \mid \Omega_C) \\
&= I\Big((S^{(1)}, S^{(2)}, S) \\
&\qquad \wedge (R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n]\setminus C,C}, V_i) \mid S\Big)
\end{aligned}$$

[†]The symbol $\perp$ means no input is given.

$$= H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n]\setminus C,C}, V_i \mid S) - H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n]\setminus C,C}, V_i \mid S^{(1)}, S^{(2)}, S). \tag{37}$$

The evaluation of the first term of (37) is as follows:

$$\begin{aligned} &H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n]\setminus C,C}, V_i, S) \\ &\overset{(a)}{=} H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{C,C}, W_{[n]\setminus C,C}, V_i, S) \\ &= H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n],C}, V_i, S) \\ &\overset{(b)}{=} H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n],C}, V_C, V_i, S) \\ &\overset{(c)}{=} H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n],C}, V_C, S) \\ &\overset{(d)}{=} H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n]\setminus C,C}, S) \\ &\le H(R_{C,[t]}) + H(V_C^{(1)}) + H(V_C^{(2)}) \\ &\quad + H(W_{[n]\setminus C,C}) + H(S), \end{aligned} \tag{38}$$

where the last inequality holds from the subadditivity of the entropy (Prop. 2.1–2.), and the marked equalities follow from the following observations with Prop. 2.1–1.

(a) From (15) and (17), $W_{i,C}$ can be uniquely determined from $V_i^{(1)}, V_i^{(2)}$ and $R_{i,[t]}$ for all $i \in C$. Hence we can include $W_{C,C}$.
(b) From (22), V_i can be uniquely computed from $W_{[n],i}$ for all $i \in C$. Hence, we can include V_C.
(c) Using Prop. 3.3 for the relation (23), V_C and S determines V_i uniquely. Hence, we can exclude V_i.
(d) We can exclude $W_{C,C}$ and V_C due to the observations (a) and (b) above, respectively.

Noticing that $|C| = t$, we have the following cardinality bounds:

$$\begin{aligned} H(R_{C,[t]}) &= t^2 \log|\mathbb{F}|, \\ H(V_C^{(1)}) &\le t \log|\mathbb{F}|, \\ H(V_C^{(2)}) &\le t \log|\mathbb{F}|, \\ H(W_{[n]\setminus C,C}) &\le t(n-t) \log|\mathbb{F}|. \end{aligned}$$

From these (in)equalities, (38) can be bounded as

$$\begin{aligned} &H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n]\setminus C,C}, V_i, S) \\ &\quad \le t(n+2) \log|\mathbb{F}| + H(S). \end{aligned}$$

Hence, we obtain

$$\begin{aligned} &H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n],C}, V_i \mid S) \\ &\quad \le t(n+2) \log|\mathbb{F}|. \end{aligned} \tag{39}$$

To evaluate the second term in (37), we compute:

$$\begin{aligned} &H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n]\setminus C,C}, V_i, S^{(1)}, S^{(2)}, S) \\ &\overset{(e)}{=} H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W_{[n]\setminus C,C}, S^{(1)}, S^{(2)}, S) \\ &= H(R_{C,[t]}, V_C^{(1)}, S^{(1)}, V_C^{(2)}, S^{(2)}, W_{[n]\setminus C,C}, S) \\ &\overset{(f)}{=} H(R_{C,[t]}, R_{[t]}^{(1)}, S^{(1)}, R_{[t]}^{(2)}, S^{(2)}, W_{[n]\setminus C,C}, S) \\ &\overset{(g)}{=} H(R_{C,[t]}, R_{[t]}^{(1)}, S^{(1)}, R_{[t]}^{(2)}, S^{(2)}, W_{[n]\setminus C}, W_{[n]\setminus C,C}, S) \\ &\overset{(h)}{=} H(R_{C,[t]}, R_{[t]}^{(1)}, S^{(1)}, R_{[t]}^{(2)}, S^{(2)}, W_{[n]\setminus C}, R_{[n]\setminus C,[t]}, S) \\ &= H(R_{[t]}^{(1)}, S^{(1)}, R_{[t]}^{(2)}, S^{(2)}, R_{[n],[t]}, W_{[n]\setminus C}, S) \\ &\overset{(i)}{=} H(R_{[t]}^{(1)}, S^{(1)}, R_{[t]}^{(2)}, S^{(2)}, R_{[n],[t]}, S) \\ &\overset{(j)}{=} H(R_{[t]}^{(1)}) + H(R_{[t]}^{(2)}) + H(R_{[n],[t]}) + H(S^{(1)}, S^{(2)}, S) \\ &\overset{(k)}{=} t(n+2) \log|\mathbb{F}| + H(S^{(1)}, S^{(2)}, S), \end{aligned} \tag{40}$$

where the marked equalities follow from the following reasons:

(e) V_i can be excluded due to the same reasons as (a)–(d) in (38).
(f) Since (12) holds in π^{mult}, we can replace $V_C^{(i)}$ with $R_{[t]}^{(i)}$ for $i = 1, 2$.
(g) Recall that we compute $V_{[n]}^{(i)}$ from $(R_{[t]}^{(i)}, S^{(i)})$ for $i = 1, 2$. Then, from (15), we have $W_{[n]}$ from $V_{[n]}^{(1)}$ and $V_{[n]}^{(2)}$, which involves $W_{[n]\setminus C}$. Hence, we can add $W_{[n]\setminus C}$ by considering Prop. 2.1–1.
(h) From (19), we can replace $W_{[n]\setminus C,C}$ with $R_{[n]\setminus C,[t]}$.
(i) Since $S^{(1)}$, $S^{(2)}$, $R_{[t]}^{(1)}$, $R_{[t]}^{(2)}$, and $R_{[n],[t]}$, are all inputs and randomness used in π^{mult}, $W_{[n]\setminus C}$ can be computed from them. Hence, considering Prop. 2.1–1., $W_{[n]\setminus C}$ can be excluded.
(j) Random variables $R_{[t]}^{(1)}$, $R_{[t]}^{(2)}$, $R_{[n],[t]}$, and $(S^{(1)}, S^{(2)}, S)$ are mutually independent from the protocol π^{mult}.
(k) It is easy to check that $H(R_{[t]}^{(1)}) = t \log|\mathbb{F}|$, $H(R_{[t]}^{(2)}) = t \log|\mathbb{F}|$, and $H(R_{[n],[t]}) = nt \log|\mathbb{F}|$.

Hence, we have

$$\begin{aligned} &H(R_{C,[t]}, V_C^{(1)}, V_C^{(2)}, W(C), V_i \mid S^{(1)}, S^{(2)}, S) \\ &\quad = t(n+2) \log|\mathbb{F}|, \end{aligned} \tag{41}$$

Combining (39) and (41), (37) is upper-bounded as follows:

$$I(\Omega_{\overline{C}} \wedge \Phi_C^{\mathrm{mult}} \mid \Omega_C) \le 0.$$

Since the conditional mutual entropy is non-negative, we obtain (36). □

7. Security Proofs via Sufficient Statistics

In the previous section, we proved that the addition and multiplication protocols are t-private in the sense of Theorem 5.6–(iv), i.e., from the view of information measures. In this section, we prove (v) in Theorem 5.6 for these protocols.

To prove (v) in Theorem 5.6, we review Fisher's factorization theorem. We omit the proof, but the readers can refer to [16].

Proposition 7.1 (Fisher's Factorization Theorem, [16])**.** A statistic $\sigma(X)$ is sufficient with respect to θ if and only if $P_{X|\boldsymbol{\Theta}}(x|\theta) = g_\theta(\sigma(x))h(x)$ holds where the functions $g_\theta(\cdot)$ and $h(\cdot)$ satisfy

- $g_\theta(\cdot)$ is a function that depends on θ, and
- $h(\cdot)$ is a function that does not depend on θ. □

Applying Proposition 7.1 to (v) in Theorem 5.6, we immediately obtain the following corollary:

Corollary 7.2 (Factorization in MPC)**.** An MPC is t-secure if Ω_C is a sufficient statistic with respect to $\omega_{\overline{C}}$, which holds if and only if

$$P_{\boldsymbol{\Phi}_C|\Omega_{\overline{C}}}(\varphi_C|\omega_{\overline{C}}) = g_{\omega_{\overline{C}}}(\omega_C)h(\varphi_C), \tag{42}$$

holds where $g_{\omega_{\overline{C}}}(\cdot)$ and $h(\cdot)$ satisfy:

- $g_{\omega_{\overline{C}}}(\cdot)$ is a function that depends on $\omega_{\overline{C}}$.
- $h(\cdot)$ is a function that does not depend on $\omega_{\overline{C}}$.

Utilizing Corollary 7.2, we can prove the following theorem:

Theorem 7.3. The addition protocol π^{add} and the multiplication protocol π^{mult} are t-private in the sense of (v) in Theorem 5.6.

In the following, we verify (42) for π^{add} and π^{mult}.

7.1 Security Proofs on π^{add}

Proof of Theorem 7.3 on π^{add}. Recall the following actual values of inputs, outputs, and views.

$$\begin{aligned}
\omega_C &\coloneqq (\perp, s)\\
\omega_{\overline{C}} &\coloneqq \left((s^{(1)}, s^{(2)}), s\right)\\
\varphi_C^{\mathrm{add}} &\coloneqq \left(\perp; v_C^{(1)}, v_C^{(2)}, w_i\right).
\end{aligned}$$

In order to check the factorization theorem, we calculate[†]

$$P_{\boldsymbol{\Phi}_C^{\mathrm{add}}\Omega_{\overline{C}}}(\varphi_C^{\mathrm{add}}, \omega_{\overline{C}}) = P(v_C^{(1)}, v_C^{(2)}, w_i, s^{(1)}, s^{(2)}, s).$$

We observe that s and w_i are uniquely determined by $(s^{(1)}, s^{(2)})$ and $(v_C^{(1)}, v_C^{(2)}, s)$, respectively, in the following manners.

- $s = s^{(1)} + s^{(2)}$.
- From (13), we can compute $w_C = v_C^{(1)} + v_C^{(2)}$. Then, recalling that (14), we can compute w_i from (w_C, s). Namely, we have a map

$$\mu_{C,i}^t : (v_C^{(1)} + v_C^{(2)}, s) \mapsto w_i.$$

Hence, we have

$$\begin{aligned}
&P(v_C^{(1)}, v_C^{(2)}, w_i, s^{(1)}, s^{(2)}, s)\\
&= P(v_C^{(1)}, v_C^{(2)}, s^{(1)}, s^{(2)})\\
&\times \mathbb{1}\left(\mu_{C,i}^t(v_C^{(1)} + v_C^{(2)}, s) = w_i\right)\mathbb{1}(s = s^{(1)} + s^{(2)}),
\end{aligned}$$

where $\mathbb{1}(\cdot)$ is an indicator function that takes 1 when the relation in the parenthesis holds; otherwise, it takes 0.

The probability $P(v_C^{(1)}, v_C^{(2)}, s^{(1)}, s^{(2)})$ above can be transformed into

$$\begin{aligned}
P(v_C^{(1)}, s^{(1)}, v_C^{(2)}, s^{(2)}) &= P(r_{[t]}^{(1)*}, s^{(1)}, r_{[t]}^{(2)*}, s^{(2)})\\
&= \frac{P(s^{(1)}, s^{(2)})}{|\mathbb{F}|^{2t}},
\end{aligned}$$

Summarizing the above, we have

$$\begin{aligned}
&P_{\boldsymbol{\Phi}_C^{\mathrm{add}}, \Omega_{\overline{C}}}(\varphi_C^{\mathrm{add}}, \omega_{\overline{C}}) = \frac{P(s^{(1)}, s^{(2)})}{|\mathbb{F}|^{2t}}\\
&\times \mathbb{1}\left(\mu_{C,i}^t(v_C^{(1)} + v_C^{(2)}, s) = w_i\right)\mathbb{1}(s = s^{(1)} + s^{(2)}),
\end{aligned}$$

which yield[††]

$$\begin{aligned}
&P_{\boldsymbol{\Phi}_C^{\mathrm{add}}|\Omega_{\overline{C}}}(\varphi_C^{\mathrm{add}} \mid \omega_{\overline{C}})\\
&= \frac{1}{|\mathbb{F}|^{2t}}\mathbb{1}\left(\mu_{C,i}^t(v_C^{(1)} + v_C^{(2)}, s) = w_i\right)\mathbb{1}(s = s^{(1)} + s^{(2)}).
\end{aligned} \tag{43}$$

Therefore, we have the following decomposition such that

$$\begin{aligned}
g_{\omega_{\overline{C}}}(\omega_C) &= \mathbb{1}(s = s^{(1)} + s^{(2)}),\\
h(\varphi_C^{\mathrm{add}}) &= \frac{1}{|\mathbb{F}|^{2t}}\mathbb{1}\left(\mu_{C,i}^t(v_C^{(1)} + v_C^{(2)}, s) = w_i\right).
\end{aligned}$$

Recalling the protocol π^{add}, s is recovered from $w_C = v_C^{(1)} + v_C^{(2)}$ and w_i. Hence, we can see that $h(\varphi_C^{\mathrm{add}})$ depends neither $s^{(1)}$ nor $s^{(2)}$, which completes the proof. □

7.2 Security Proofs on π^{add}

Proof of Theorem 7.3 on π^{mult}. Recall the following actual values of inputs, outputs, and views.

$$\begin{aligned}
\omega_C &\coloneqq (\perp, s)\\
\omega_{\overline{C}} &\coloneqq ((s^{(1)}, s^{(2)}), s)\\
\varphi_C^{\mathrm{mult}} &\coloneqq (\perp, r_C^t; v_C^{(1)}, v_C^{(2)}, w_{[n]\setminus C,[n]}, v_i).
\end{aligned}$$

In a similar way with the proof on π^{add}, we compute

$$P_{\boldsymbol{\Phi}_C^{\mathrm{mult}}\Omega_{\overline{C}}}(\varphi_C^{\mathrm{mult}}, \omega_{\overline{C}})$$

[†]The random variables in the suffix, i.e., X of $P_X(\cdot)$, are sometimes omitted due to the space limitation hereafter.

[††]Note that $P_{\boldsymbol{\Phi}_C^{\mathrm{add}}|\Omega_{\overline{C}}}(\varphi_C^{\mathrm{add}} \mid \omega_{\overline{C}})$ can take an arbitrary value when s is not consistent with $s^{(1)}$ and $s^{(2)}$. Hence, to avoid such arbitrariness, we define it to be 0 when $s \neq s^{(1)} + s^{(2)}$ in (43) to indicate that the conditional probability is meaningless. The same discussion will apply to the case of π^{mult} when $s \neq s^{(1)}s^{(2)}$ in (44).

$$= P(v_C^{(1)}, s^{(1)}, v_C^{(2)}, s^{(2)}, r_{C,[t]}, w_{[n]\setminus C,C}, v_i, s)$$

We first point out that v_i is uniquely determined by $v_i^{(1)}, v_i^{(2)}, r_{i,[t]}$ and s for all $i \in C$ with the same reason with (a) and (b) in (38). Hence, we define the map μ_C such that

$$\mu_C : (v_C^{(1)}, v_C^{(2)}, r_{C,[t]}, s) \mapsto v_i.$$

We also observe that s is determined uniquely by $s^{(1)}$ and $s^{(2)}$. Hence, we have

$$\begin{aligned}
&P(v_C^{(1)}, s^{(1)}, v_C^{(2)}, s^{(2)}, r_{C,[t]}, w_{[n]\setminus C,C}, v_i, s)\\
&= P(v_C^{(1)}, s^{(1)}, v_C^{(2)}, s^{(2)}, r_{C,[t]}, w_{[n]\setminus C,C})\\
&\times \mathbb{1}(s = s^{(1)}s^{(2)})\mathbb{1}\left(\mu(v_C^{(1)}, v_C^{(2)}, r_{C,[t]}, s) = v_i\right).
\end{aligned}$$

The probability $P(v_C^{(1)}, s^{(1)}, v_C^{(2)}, s^{(2)}, r_{C,[t]}, w_{[n]\setminus C,C})$ can be transformed as

$$\begin{aligned}
&P(v_C^{(1)}, s^{(1)}, v_C^{(2)}, s^{(2)}, r_{C,[t]}, w_{[n]\setminus C,C})\\
&\overset{(a)}{=} P(r_{[t]}^{(1)*}, s^{(1)}, r_{[t]}^{(2)*}, s^{(2)}, r_{C,[t]}, w_{[n]\setminus C,C})\\
&\overset{(b)}{=} P(r_{[t]}^{(1)*}, s^{(1)}, r_{[t]}^{(2)*}, s^{(2)}, r_{C,[t]}, w_{[n]\setminus C}, w_{[n]\setminus C,C})\\
&\overset{(c)}{=} P(r_{[t]}^{(1)*}, s^{(1)}, r_{[t]}^{(2)*}, s^{(2)}, r_{C,[t]}, w_{[n]\setminus C}, r^*_{[n]\setminus C,[t]})\\
&= P(r_{[t]}^{(1)*}, r_{[t]}^{(2)*}, r_{C,[t]}, r^*_{[n]\setminus C,[t]})P(s^{(1)}, s^{(2)})\\
&= \frac{P(s^{(1)}, s^{(2)})}{|\mathbb{F}|^{t(n+2)}},
\end{aligned}$$

where the marked equalities (a), (b), and (c) hold with the same reason with (f), (g), and (h) in (40), respectively. The values $r_{[t]}^{(1)*}$, $r_{[t]}^{(2)*}$, and $r^*_{[n]\setminus C,[t]}$ are the random numbers uniquely determined in each transformation.

Summarizing, we have

$$\begin{aligned}
&P_{\boldsymbol{\Phi}_C^{\mathrm{mult}},\Omega_{\overline{C}}}(\varphi_C^{\mathrm{mult}}, \omega_{\overline{C}}) = \frac{P(s^{(1)}, s^{(2)})}{|\mathbb{F}|^{t(n+2)}}\\
&\times \mathbb{1}(s = s^{(1)}s^{(2)})\mathbb{1}\left(\mu(v_C^{(1)}, v_C^{(2)}, r_{C,[t]}, s) = v_i\right).
\end{aligned}$$

Hence, It holds that

$$\begin{aligned}
&P_{\boldsymbol{\Phi}_C^{\mathrm{mult}}|\Omega_{\overline{C}}}(\varphi_C^{\mathrm{mult}} \mid \omega_{\overline{C}}) = \frac{1}{|\mathbb{F}|^{t(n+2)}}\\
&\times \mathbb{1}(s = s^{(1)}s^{(2)})\mathbb{1}\left(\mu(v_C^{(1)}, v_C^{(2)}, r_{C,[t]}, s) = v_i\right), \qquad (44)
\end{aligned}$$

which is decomposed as

$$\begin{aligned}
&g_{\omega_{\overline{C}}}(\omega_C) = \mathbb{1}(s = s^{(1)}s^{(2)}),\\
&h(\varphi_C^{\mathrm{mult}}) = \frac{1}{|\mathbb{F}|^{t(n+2)}}\mathbb{1}\left(\mu(v_C^{(1)}, v_C^{(2)}, r_{C,[t]}, s) = v_i\right).
\end{aligned}$$

Recalling the protocol π^{mult}, s is recovered from $\varphi_C^{\mathrm{mult}}$, which contains neither $s^{(1)}$ nor $s^{(2)}$, which completes the proof. □

8. Concluding Remarks

This paper explored the simulation-based security of MPC under a semi-honest setting through a lens of information theory and statistics. For this purpose, we reviewed secret sharing with Shamir's scheme, the addition and multiplication protocols of BGW protocols, denoted by π^{add} and π^{mult}, respectively.

In order to understand the simulation-based security of MPC from information theory and statistics, we introduced simulation-based security for information-theoretic MPC in a standard manner [13], [14], and we discussed several equivalent formalizations of simulation-based security for MPC protocols. We obtained four equivalent formalizations of simulation-based security based on conditional probabilities, Markov chains, conditional mutual information, and sufficient statistics.

Instead of omitting the proof of simulation-based security, i.e., constructions of simulators [13], [14], we showed two types of security proofs for BGW protocols for addition and multiplication in a semi-honest model. One proof is based on information measures, and the other on sufficient statistics. The proofs based on information measures exploited the same techniques as the security proof of secret-sharing schemes, i.e., random variables' inclusion, deletion, and replacement. The key in the proofs based on sufficient statistics was the factorization theorem, which suggested calculating the conditional probability distributions of views given the pair of inputs and outputs. This was achieved by the observations in the proofs based on information measures. Both proofs seemed not to use simulation explicitly.

For future work, we list the problems not investigated in this paper.

- There are a lot of security proofs based on simulations, e.g., in [5]. Applying the techniques in this paper to these security proofs is worth investigating. In particular, it would be interesting if we could capture the security formalization of malicious security by our approaches. Note that [13] proved the malicious security of BGW protocols when $t < n/3$ with the aid of verifiable secret-sharing schemes [32].
- By using the formalization by information measures, there is a possibility that we can allow information leakage for MPC like ramp secret sharing [33], [34].
- From the computational security side, information measures in computational complexity theory are also discussed in, for instance, [8], [35], [36]. Can we prove the security of computationally secure MPCs using these information measures? Furthermore, it would be interesting if we could define a computational-theoretic version of sufficient statistics.

Acknowledgments

The author would like to thank Yuichi Kaji for inviting the au-

thor to submit the paper to the special issue. He is also grateful to the reviewers for their insightful comments. The author's work is supported by JSPS KAKENHI Grant Numbers JP21H03395, JP22H03590, JP23H00468, JP23H00479, and JP23K17455.

References

[1] C. Shannon, "A mathematical theory of communication," Bell Syst. Tech. J., vol.27, pp.379–423, July and Oct. 1948.
[2] W. Diffie and M. Hellman, "New directions in cryptography," IEEE Trans. Inf. Theory, vol.22, no.6, pp.644–654, 1976.
[3] R.L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Commun. ACM, vol.21, no.2, pp.120–126, 1978.
[4] S. Goldwasser and S. Micali, "Probabilistic encryption," J. Comput. Syst. Sci., vol.28, no.2, pp.270–299, 1984.
[5] Y. Lindell, "How to simulate it — A tutorial on the simulation proof technique," Tutorials on the Foundations of Cryptography, pp.277–346, 2017.
[6] A. Russell and H. Wang, "How to foll an unbounded adversary with a short key," IEEE Trans. Inf. Theory, pp.1330–1140, 2006. Preliminary version: EUROCRYPT 2002, LNCS 2332, Springer–Verlag, pp.133–148, 2002.
[7] M. Iwamoto, K. Ohta, and J. Shikata, "Security formalizations and their relationships for encryption and key agreement in information-theoretic cryptography," IEEE Trans. Inf. Theory, vol.64, no.1, pp.654–685, 2018.
[8] S.P. Vadhan, "Computational entropy," Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali, pp.693–726, ACM Books, 2019.
[9] C.E. Shannon, "Communication theory of secrecy systems," Bell Tech. J., vol.28, no.4, pp.656–715, Oct. 1949.
[10] G.R. Blakley, "Safeguarding cryptographic keys," AFIPS 1979 National Computer Conference, vol.48, pp.313–317, 1979.
[11] A. Shamir, "How to share a secret," Commun. ACM, vol.22, no.11, pp.612–613, 1979.
[12] M. Ben-Or, S. Goldwasser, and A. Wigderson, "Completeness theorems for non-cryptographic fault tolerant distributed computation," Proc. 20th Annual ACM Sym. Theory of Computing (STOC88), pp.1–10, 1988.
[13] G. Asharov and Y. Lindell, "A full proof of the bgw protocol for perfectly secure multiparty computation," J. Cryptol., vol.30, no.1, pp.58–151, 2017.
[14] R. Cramer, I. Damgård, and J.B. Nielsen, Secure Multiparty Computation and Secret Sharing, Cambridge University Press, 2015.
[15] E.D. Karnin, J.W. Greene, and M.E. Hellman, "On secret sharing systems," IEEE Trans. Inf. Theory, vol.29, no.1, pp.35–41, 1983.
[16] E.L. Lehmann and J.P. Romano, Testing Statistical Hypotheses, 3rd ed., Springer Texts in Statistics, Springer, 2008.
[17] T.M. Cover and J.A. Thomas, Elements of Information Theory, 2nd ed., Wiley and Interscience, 2006.
[18] M. Itoh, A. Saito, and T. Nishizeki, "Secret sharing scheme realizing general access structure," IEEE GLOBECOM, pp.99–102, 1987.
[19] M. Itoh, A. Saito, and T. Nishizeki, "Multiple assignment scheme for sharing secret," J. of Cryptology, vol.6, pp.15–20, 1993. Preliminary version: IEEE GLOBECOM'87, pp.99–102, 1987.
[20] J. Benaloh and J. Leichter, "Generalized secret sharing and monotone functions," Advances in Cryptology — CRYPTO'88, LNCS 403, pp.27–35, Springer-Verlag, 1990.
[21] H. Krawczyk, "Secret sharing made short," Advances in Cryptology — CRYPTO'93, D.R. Stinson, ed., Berlin, Heidelberg, pp.136–146, Springer Berlin Heidelberg, 1994.
[22] A. Shamir, R. Rivest, and L. Adleman, "Mental poker," Technical Report, Technical Memo LCS/TM-125, Massachusetts Institute of Technology, 1979.
[23] A. Yao, "Protocols for secure computations," 23rd Annual Symposium on Foundations of Computer Science, pp.160–164, 1982.
[24] O. Goldreich, S. Micali, and A. Wigderson, "Proofs that yield nothing but their validity and a methodology of cryptographic protocol design," 27th Annual Symposium on Foundations of Computer Science (sfcs 1986), pp.174–187, 1986.
[25] A.C.C. Yao, "How to generate and exchange secrets," 27th Annual Symposium on Foundations of Computer Science (sfcs 1986), pp.162–167, IEEE, 1986.
[26] M.O. Rabin, "How to exchange secrets with oblivious transfer," Technical Report, Harvard University, Cryptology ePrint Archive, Paper 2005/187, 1981.
[27] D. Evans, V. Kolesnikov, and M. Rosulek, A Pragmatic Introduction to Secure Multi-Party Computation, NOW Publishers, 2018.
[28] D. Chaum, C. Crépeau, and I. Damgard, "Multiparty unconditionally secure protocols," Proc. Twentieth Annual ACM Symposium on Theory of Computing, STOC'88, New York, NY, USA, pp.11–19, Association for Computing Machinery, 1988.
[29] R. Gennaro, M. Rabin, and T. Rabin, "Simplified vss and fast-track multiparty computations with applications to threshold cryptography," 17th Annual ACM Symposium on Principles of Distributed Computing (PODC), pp.101–111, ACM, 1998.
[30] O. Goldreich, Foundations of Cryptography Volume II: Basic Applications, Cambridge University Press, 2004.
[31] C. Blundo, A. De Santis, G. Persiano, and U. Vaccaro, "Randomness complexity of private computation," Comput. Complex., vol.8, no.2, pp.145–168, 1999.
[32] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch, "Verifiable secret sharing and achieving simultaneity in the presence of faults," 26th Annual Symposium on Foundations of Computer Science (sfcs 1985), pp.383–395, 1985.
[33] G.R. Blakley and C. Meadows, "Security of ramp schemes," Advances in Cryptology–CRYPTO'84, LNCS 196, pp.242–269, Springer-Verlag, 1985.
[34] H. Yamamoto, "Secret sharing system using (k, L, n) threshold scheme," Electronics and Communications in Japan (Part I: Communications), vol.69, no.9, pp.46–54, 1986.
[35] I. Haitner and S. Vadhan, "The many entropies in one-way functions," Tutorials on the Foundations of Cryptography: Dedicated to Oded Goldreich, pp.159–217, Information Security and Cryptography, Springer International Publishing, 2017.
[36] R. Agrawal, Y.H. Chen, T. Horel, and S. Vadhan, "Unifying computational entropies via Kullback–Leibler divergence," Advances in Cryptology — CRYPTO 2019, A. Boldyreva and D. Micciancio, eds., pp.831–858, Springer International Publishing, 2019.

Mitsugu Iwamoto received the B.E., M.E., and Ph.D. degrees from the University of Tokyo, Tokyo, Japan, in 1999, 2001, and 2004, respectively. In 2004, he joined the University of Electro-Communications, where he is currently a Professor in the Department of Informatics. His research interests include information theory, information security, and cryptography. He is a member of IEICE, IEEE, and IACR.