Game-Theoretic Security for Two-Party Protocols[*]

Haruna Higo[†] Keisuke Tanaka[‡] Akihiro Yamada[§] Kenji Yasunaga[¶]

November 16, 2016

Abstract

Asharov, Canetti, and Hazay (Eurocrypt 2011) studied how game-theoretic concepts can be used to capture the cryptographic properties of correctness, privacy, and fairness in two-party protocols for fail-stop adversaries. In this work, we further study the characterization of the cryptographic properties of specific two-party protocols, oblivious transfer (OT) and commitment, in terms of game theory. Specifically, for each protocol, OT and commitment, we define a two-party game between rational sender and receiver together with their utility functions. Then, we prove that a given protocol satisfies cryptographic properties if and only if the strategy of following the protocol is in a Nash equilibrium. Compared to the previous work of Asharov et al., our characterization has several advantages: The game is played by multiple rational parties; All the cryptographic properties of OT/commitment are characterized by a single game; Security for malicious adversaries is considered; Utility functions are specified in general forms based on the preferences of the parties; A solution concept employed is a plain Nash equilibrium.

1 Introduction

In cryptography, two-party protocols are designed for two parties to compute some function while concealing the input from each other. To guarantee the secrecy of the inputs, we consider the case where one of the parties is an adversary who is interested in attacking the other, e.g., digging out the other's secret. In general, the adversary acts only for attacking the other, and does not care about protecting his own secret. Also, cryptography only considers the situations where at least one party is honest, i.e., always follows the protocol description.

Game theory mathematically analyzes decision making of multiple parties. In particular, non-cooperative game theory deals with the situations where the parties act independently. The parties are said to be rational, and they only care about their own preferences to achieve their best satisfactions. If a party has two or more preferences, he considers the trade-offs among them and aims to obtain the most reasonable result.

As described, both non-cooperative game theory and cryptography study the situations where parties act. However, they capture such situations from different perspectives. By assessing the situations realistically,

[*] Preliminary results appeared at the 17th Australasian Conference on Information Security and Privacy (ACISP 2012) [16] and the 8th International Workshop on Security (IWSEC 2013) [17].

[†] NEC Corporation, Japan. Email: `h-higo@aj.jp.nec.com`. Most of the work was done while the author was a student at Tokyo Institute of Technology.

[‡] Tokyo Institute of Technology, Japan. Email: `keisuke@is.titech.ac.jp`.

[§] This work was done while the author was a student at Tokyo Institute of Technology.

[¶] Kanazawa University, Japan. Email: `yasunaga@se.kanazawa-u.ac.jp`.

even adversaries may be reluctant to reveal their secrets. Also, if a party is sure that there is no danger, he may deviate from the protocol description to obtain more information than following it. That is, all parties may not be completely honest. In a game-theoretic framework, it seems to be possible to characterize two-party protocols in such a realistic perspective.

There is a line of work using game-theoretic concepts to study cryptographic protocols. For a survey on the joint work of cryptography and game theory, we refer to [18, 4]. Halpern and Teague [14] introduced such approach of study on secret sharing. They study it in the presence of rational parties, seeking for secure protocols in a game-theoretic framework. Their work has been followed in many subsequent works, and the field is called rational secret sharing (see [5] and the references therein for the subsequent works). Besides secret sharing, there are several studies using game-theoretic frameworks for cryptographic protocols, e.g., two-party computation [2, 12], leader election [10, 1], Byzantine agreement [13], public-key encryption [21], and protocol design [6, 7].

Asharov, Canetti, and Hazay [2] studied how game-theoretic concepts can be used to capture the three requirements of the two-party protocols in cryptography, correctness, privacy, and fairness. They characterize these requirements *individually* by using a game-theoretic concept, a *computational Nash equilibrium*. They focus on two-party protocols in the *fail-stop model*, in which adversaries are allowed to choose whether to abort or continue at each round, but cannot conduct other actions such as sending illegal messages. Using game-theoretic concepts, they characterize the requirements of two-party protocols in the following way: A protocol satisfies a "certain" requirement if and only if the strategy of honestly following the description of the protocol is in a computational Nash equilibrium in a "certain" game defined with "certain" utility functions. For privacy and correctness, they showed the equivalence between the corresponding cryptographic and the game-theoretic definitions. For fairness, they showed that their game-theoretic definition is strictly weaker than existing cryptographic ones, and proposed a new cryptographic definition that is equivalent to the game-theoretic one. Groce and Katz [12] continued their consideration on fairness, and showed a way to circumvent impossibility results in the study of [2]. Goto and Shikata [9] studied oblivious transfer protocols with game-theoretic security[1] and universal composability.

1.1 This Work

Based on the work of Asharov et al. [2], we further explore how the cryptographic requirements can be captured in a game-theoretic framework. In particular, our target protocols are *oblivious transfer (OT)* and *commitment*.

Oblivious transfer. OT is a two-party protocol run between the sender and the receiver. The sender has two secrets x_0 and x_1, and the receiver has a choice bit $c \in \{0,1\}$. After running the protocol, the receiver obtains x_c, while the sender obtains nothing. Because of a technical reason, we restrict our attention to *two-message* OT in which the receiver sends the first message to the sender, and the sender replies with the second message to the receiver who then learns the secret x_c.

We usually consider three requirements in OT, *the sender's privacy*, *the receiver's privacy*, and *correctness*. By the sender's/receiver's privacy, it is guaranteed that the receiver/sender cannot learn anything about x_{1-c}/c, respectively. Correctness guarantees that when two parties honestly follow the protocol description, the receiver learns the secret x_c Note that, in the indistinguishability-based security definitions, the three

[1] They employ a game-theoretic characterization of our preliminary results in [16], in which we used fixed-valued utility functions and a computational Nash equilibrium.

Table 1: Summary of the characterizations of [2] and this work.

	Asharov et al. [2]	This work	
Target protocol	Two-party protocol	OT	Commitment
Adversary model	Fail-stop	Malicious	Malicious
Cryptographic properties	Correctness Privacy Fairness	Correctness Sender's privacy Receiver's privacy	Correctness Hiding Binding
# of rational parties in one game	1	2	2
# of properties captured by one game	1	3	3
Utility function	Fixed	General	General
Solution concept	Computational NE	NE	NE

requirements are defined *separately*. Thus, for example, we usually do not consider an adversary who tries to break the other party's privacy and protect its own privacy simultaneously.

Commitment. Commitment is also a two-party protocol run between the sender and the receiver. The protocol consists of two phases. In the first phase, called the *commit phase*, the sender who has a string $x \in \{0,1\}^t$ interacts with the receiver. After that, the receiver obtains a commitment string c, and the sender obtains a decommitment string d. In the latter phase, called the *open phase*, the sender persuade the receiver that the committed string is x through an interaction by using d. Finally, the receiver claims whether she accepts that x is the committed string.

We usually consider three requirements in commitment, *hiding*, *binding*, and *correctness*. Hiding is the property that the receiver cannot learn anything about the committed string x before starting the open phase. Binding is that the sender cannot generate two decommitment strings to open the commitment to two distinct strings x and x'. Correctness guarantees that when two parties honestly follow the protocol description, the receiver learns the string that was committed by the sender in the commit phase. As in the case of OT, in cryptography, each of the three properties is defined individually. Thus, we usually do not consider a party who tries to break hiding and protect binding simultaneously.

Game-theoretic characterizations. In this work, for each protocol, OT and commitment, we define a game together with the utility functions of the sender and the receiver. Then, we show that, given a protocol for OT/commitment, the strategy of honestly following the protocol is in a Nash equilibrium in this game *if and only if* the protocol satisfies *all* the cryptographic properties of OT/commitment in the *malicious model*. In other words, we present a novel way to capture the standard cryptographic security in terms of game theory. The equivalence implies that the standard security that is defined between an honest party and a malicious party is also reasonable for rational parties.

Our characterization of OT and commitment has the following advantages compared to the work of [2] (See Table 1 for the summary.):

- The game defined in our work is played between two rational parties, while every game defined in [2] is played by a single rational party. For example, the game for the privacy of party 1 in [2] is essentially played between a rational party and an honest party. Since game-theoretic concepts are of significant meaning in the presence of multiple rational parties, it is preferable to characterize a single game that is essentially played between two rational parties.

- We put multiple preferences of the parties into a single game, while each of them is characterized by different games in [2]. This means that rational parties pay attention to the trade-offs among the preferences in the game, while such trade-offs are not considered in the standard cryptographic security. Although we show the equivalence of the game-theoretic characterization of the protocols to the existing cryptographic security (therefore we call the characterization "game-theoretic security"), the strength of the game-theoretic security can be altered by considering different utility functions and solution concepts.

  Indeed, our game-theoretic characterization reveals the difference between the parties' preferences for correctness in OT and commitment. For OT, the cryptographic security is equivalent to the game-theoretic one as long as at least one of the parties has a preference for correctness. In contrast, for commitment, the game-theoretic security becomes weaker if the sender has a preference for correctness. The equivalence holds when only the receiver has a preference for correctness.

- We can capture the setting of malicious adversaries, who can take any action in the protocol. The malicious model is stronger and more realistic than the fail-stop model that is studied in [2], where adversaries choose to "continue" or "stop" in each round.

- Utility functions are specified in general forms based on the preferences of the parties. In [2], utility functions are defined such that they take some fixed values, say 0 and 1, depending on the outcomes of the game. We only consider the increase and decrease based on whether the preference are satisfied or not. Thus, fixed-valued utility functions can be seen as a special case of our utility functions.

- We can capture the cryptographic requirements by plain Nash equilibrium, not *computational* Nash equilibrium. This can be done by reforming the way of perceiving the preferences. First, we define the preferences of the parties not over the outcome of a single execution, but over the algorithms used by the parties. This way of defining preferences seems natural since protocols are usually designed for the repeated use, and thus the users are not just interested in a good outcome of a single game but prefer to use a good algorithm (protocol) for multiple games. Second, we exclude from strategies sub-algorithms such as a distinguisher for guessing the secret. We consider best possible such sub-algorithms for given strategies. For example, a utility function of a party is defined such that the party prefers strategy A to B if there exists a distinguisher that predicts a challenge bit better when using A. Thereby, we can define strategies of the parties in a simplified form. As a result, we can characterize the cryptographic properties by plain Nash equilibrium.

As described above, our characterization clarifies the difference between OT and commitment regarding the parties' preferences for correctness. The difference is not obvious from the cryptographic security definitions. Thus, the game-theoretic characterization can be used to clarify the functionalities of protocols that have several cryptographic requirements which are defined individually.

The generality of game-theoretic formalizations is illustrated by our results. We can define a game-theoretic security that is equivalent to the existential cryptographic one. By considering various solution concepts and utility functions, we can define various levels of security for cryptographic protocols.

1.2 Organization

The rest of the paper is organized as follows. We review some concepts and definitions including two-party protocols, oblivious transfer, commitment, and game theory in Section 2. In Sections 3 and 4, we propose game-theoretic characterizations of oblivious transfer and commitment, respectively. We conclude the paper in Section 5.

2 Preliminaries

In this section, we provide some basic notions and notations.

A function $\mu : \mathbb{N} \to \mathbb{R}$ is said to be *negligible* if for any polynomial $p(\cdot)$, $\mu(n) < 1/|p(n)|$ for every sufficiently large n. We describe a negligible function as $\mathsf{negl}(\cdot)$. Throughout the paper, we denote by n the security parameter, and all the parties are assumed to run in time polynomial in n. Formally, it is assumed that each party receives 1^n as a part of input. We omit it if it is obvious from the context. For a set X, we denote by $x \overset{\$}{\leftarrow} X$ the process of choosing an element $x \in X$ uniformly at random. The empty string is denoted by ε.

A two-party protocol consists of interactive algorithms of the parties. Let us consider the case where two parties interacts using algorithms A_1 and A_2 on private inputs x_1 and x_2, respectively. In the interaction between A_1 and A_2, the view to the i-th ($i \in \{1,2\}$) party is denoted by $\mathsf{view}_{A_i(x_i)}(A_{3-i}(x_{3-i}))$, which is equal to $(x_i, r_i, m_i^1, m_i^2, \cdots)$, where r_i is the internal randomness of A_i, m_i^j is the j-th message sent from A_{3-i} to A_i. The output of the algorithm A_i after the interaction is denoted by $\mathsf{out}_{A_i(x_i)}(A_{3-i}(x_{3-i}))$.

For two functions $a, b : \mathbb{N} \to \mathbb{R}$, we write $a \preceq b$ if $a(n) \le b(n) + \mathsf{negl}(n)$ for every sufficiently large n, and $a \prec b$ if there is a polynomial $p(\cdot)$ such that $a(n) \le b(n) - 1/p(n)$ infinitely often. Also, we write $a \approx b$ if $a \preceq b$ and $a \succeq b$, and $a \not\approx b$ if either $a \prec b$ or $a \succ b$.

2.1 Cryptographic Notions

We define oblivious transfer and commitment, together with their cryptographic security notions according to [8, 3, 15].

In this work, we consider *two-message* oblivious transfer, where both the sender and the receiver send their own message only once.

Definition 1 (Two-message oblivious transfer)**.** *A* two-message oblivious transfer protocol OT *is a pair of two probabilistic polynomial-time algorithms, denoted by* $\mathsf{OT} = (S, R)$*. First, R runs on input $b \in \{0,1\}$, and outputs a message m_R and a state st. Second, S runs on input (x_0, x_1) and m_R, and outputs a message m_S. Finally, R runs on input m_S and st, and outputs a string y.*

By considering two-message oblivious transfer, we can define the indistinguishability-based security [15, Section 2.6]

Definition 2 (Security for oblivious transfer)**.** *Let* $\mathsf{OT} = (S, R)$ *be a two-message oblivious transfer protocol. We say* OT *is* cryptographically secure *if it satisfies the following three properties:*

- ***Receiver's privacy:*** *For any probabilistic polynomial-time algorithms S^* and D_S, inputs $x_0, x_1 \in \{0,1\}^*$ with $|x_0| = |x_1|$, and auxiliary input $z \in \{0,1\}^*$, it holds that*

$$\Pr[D_S(\mathsf{view}_{S^*(x_0,x_1,z)}(R(0))) = 1] \approx \Pr[D_S(\mathsf{view}_{S^*(x_0,x_1,z)}(R(1))) = 1].$$

- ***Sender's privacy:*** *For any deterministic polynomial-time algorithm R^*, probabilistic polynomial-time algorithm D_R, inputs $x_0, x_1, x \in \{0,1\}^*$ with $|x_0| = |x_1| = |x|$, $c \in \{0,1\}$, and auxiliary input $z \in \{0,1\}^*$, there exists a function* choice : $\{0,1\}^* \to \{0,1\}$ *such that*

  $$\Pr[D_R(\mathsf{view}_{R^*(c,z)}(S(X^0)), X^0, X^1) = 1] \approx \Pr[D_R(\mathsf{view}_{R^*(c,z)}(S(X^1)), X^0, X^1) = 1],$$

  where $c^* = \mathsf{choice}(R^*, c, z)$, $X^0 = (x_0, x_1)$, and $X^1 = (x_0, x)$ if $c^* = 0$, $X^1 = (x, x_1)$ otherwise.

- ***Correctness:*** *For any strings $x_0, x_1 \in \{0,1\}^*$ with $|x_0| = |x_1|$, and $c \in \{0,1\}$, it holds that*

  $$\Pr[\mathsf{out}_{R(c)}(S(x_0, x_1)) = x_c] \succeq 1.$$

The function choice in the sender's privacy determines which index the receiver's algorithm R^* chooses. Since we restrict R^* to be deterministic, it is possible to determine the index that R^* chooses on input (c, z). Note that the security against malicious receiver is not weakened by this restriction. Since R^* receives an auxiliary input z, R^* can use the "best" random coins as z.

Definition 3 (Commitment)**.** *A* commitment protocol Com *is a tuple of four probabilistic polynomial-time algorithms, denoted by* $\mathsf{Com} = ((S_C, S_O), (R_C, R_O))$. *The protocol consists of two phases:*

- *The commit phase is an interaction between S_C and R_C, where S_C receives $x \in \{0,1\}^t$ as an input. The output of the commit phase consists of the* commitment *string c and a private output d for the sender, called the* decommitment *string. Without loss of generality, we can consider c to be the transcript of the interaction between $S_C(x)$ and R_C, and d the view of S_C, including the private random coins of S_C.*

- *The open phase is an interaction between S_O and R_O, where S_O and R_O receive, as inputs, (x, d) and c, respectively. We assume that the first message of S_O explicitly contains x, which indicates that the sender is to persuade the receiver that the committed string is x. After the interaction, R_O outputs* 1 *if the receiver accepts, and* 0 *otherwise.*

Definition 4 (Security for commitment)**.** *Let* $\mathsf{Com} = ((S_C, S_O), (R_C, R_O))$ *be a commitment protocol. We say* Com *is* cryptographically secure *if it satisfies the following three properties:*

- ***Hiding:*** *For any probabilistic polynomial-time algorithms R_C^* and D, inputs $x_0, x_1 \in \{0,1\}^t$, and auxiliary input $z \in \{0,1\}^*$, it holds that*

  $$\Pr[D(\mathsf{view}_{R_C^*(z)}(S_C(x_0)), x_0, x_1) = 1] \approx \Pr[D(\mathsf{view}_{R_C^*(z)}(S_C(x_1)), x_0, x_1) = 1]$$

- ***Binding:*** *For any probabilistic polynomial-time algorithms S_C^*, S_O^*, and F, input $x \in \{0,1\}^t$, and auxiliary input $z \in \{0,1\}^*$, it holds that*

  $$\Pr[\mathsf{out}_{R_O(c)}(S_O^*(x, d, z)) = \mathsf{out}_{R_O(c)}(S_O^*(x', d', z)) = 1] \preceq 0,$$

  where c and d are the commitment and decommitment strings generated by the interaction between $S_C^*(x, z)$ and R_C, (x', d') is the output of $F(\mathsf{view}_{S_C^*(x,z)}(R_C))$, where $x' \in \{0,1\}^t \setminus \{x\}$.

- ***Correctness:*** *For any $x \in \{0,1\}^t$, it holds that*

  $$\Pr[\mathsf{out}_{R_O(c)}(S_O(x, d)) = 1] \succeq 1,$$

  where c and d are the commitment and decommitment strings generated by the interaction between $S_C(x)$ and R_C.

2.2 Game-Theoretic Notions

A strategic-form game consists of three elements: a set of parties, a set of possible strategies for the parties, and utility functions. We define a two-party game as $\Gamma = (N, (A_S, A_R), (U_S, U_R))$, where $N = \{S, R\}$ is the set of parties, A_i is a set of strategies for party $i \in N$, and U_i is the utility function for party $i \in N$. The utility function U_i maps a pair of strategies $(\sigma_S, \sigma_R) \in A_S \times A_R$ to a real number which represents preferences of party i when the game is played with the pair (σ_S, σ_R).

Solution concepts characterize which tuples of strategies are likely to be chosen by the parties. While there are many solution concepts introduced in the field of game theory, we employ *Nash equilibrium*, which is the most commonly used one. When all parties choose a strategy in a Nash equilibrium, no party gains his utility by changing his strategy unilaterally. Namely, if parties are assumed to choose a Nash equilibrium strategy, no party has any incentive to change his strategy.

Definition 5 (Nash equilibrium)**.** *Let $\Gamma = (N, (A_S, A_R), (U_S, U_R))$ be a two-party game. A tuple of their strategies (σ_S, σ_R) is in a* Nash equilibrium *in the game Γ if for every strategies $\sigma'_S \in A_S$ and $\sigma'_R \in A_R$, it holds that*

$$U_S(\sigma'_S, \sigma_R) \leq U_S(\sigma_S, \sigma_R), \text{ and } U_R(\sigma_S, \sigma'_R) \leq U_R(\sigma_S, \sigma_R).$$

3 Game-Theoretic Security for Oblivious Transfer

In this section, we characterize the security of two-message oblivious transfer in terms of game theory. We show that our game-theoretic security is equivalent to the cryptographic one. Then, we discuss the implication of the equivalence.

3.1 Definition

First, we define an experiment for the execution of an oblivious transfer protocol. By specifying natural preferences of the parties, we define a game-theoretic security for oblivious transfer. A Nash equilibrium is used as a solution concept in the security definition.

Experiment. For a two-message oblivious transfer protocol, we define an experiment between a sender and a receiver. The sender has two polynomial-time algorithms (S, D_S) as a strategy, and the receiver also has two polynomial-time algorithms (R, D_R).

In the experiment, first, bits b and c are chosen uniformly at random. Then, the sender and the receiver execute the protocol using S and R. The receiver, on input c, generates the first message m_R by using R. After that, the sender, on input a pair (x'_0, x'_1) and m_R, generates the second message m_S by using S. We assume that the receiver wants to obtain x'_c that is indicated by the choice bit. The actual input to the sender is set to be X^b, where $X^0 = (x_0, x_1)$, and $X^1 = (x_0, x)$ if $c = 0$, and $X^1 = (x, x_1)$ otherwise. After the execution, the sender tries to predict c by using D_S, and the receiver tries to predict b by using D_R. More specifically, D_S tries to guess whether the receiver's choice c is 0 or 1, and D_R does whether the other input of the sender (namely, one not chosen by the receiver) is x or x_{1-c}. We note that the receiver will obtain x_c regardless of whether the input to the sender is X^0 or X^1.

We define the experiment formally. (See also Figure 1.)

Definition 6 (Experiment for oblivious transfer)**.** *Let S, R, D_S, D_R be algorithms, $x_0, x_1, x, z_S, z_R \in \{0,1\}^*$ with $|x_0| = |x_1| = |x|$, and $b, c \in \{0,1\}$. For a function* $\mathsf{choice} : \{0,1\}^* \to \{0,1\}$*, we define*

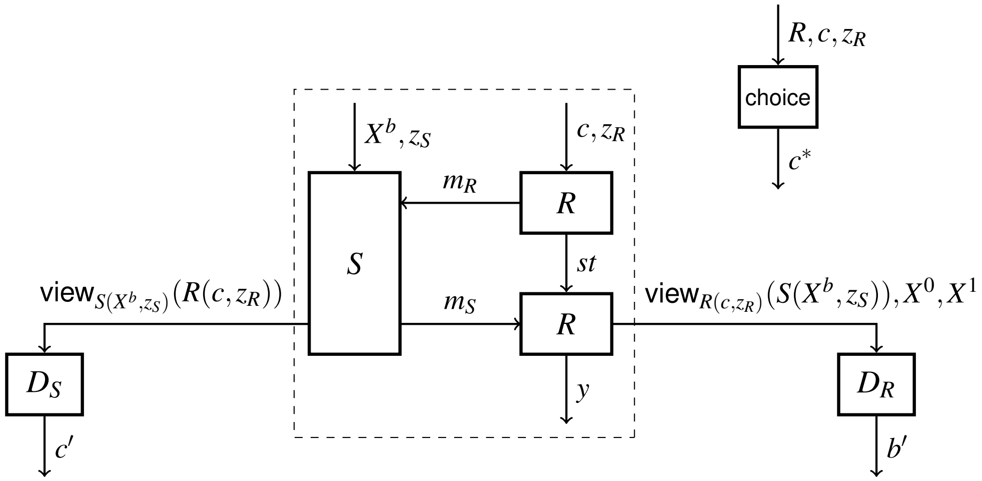
R, c, z_R
choice
c^*
X^b, z_S
c, z_R
m_R
R
S
st
$\mathsf{view}_{S(X^b,z_S)}(R(c,z_R))$
m_S
R
$\mathsf{view}_{R(c,z_R)}(S(X^b,z_S)), X^0, X^1$
D_S
y
D_R
c'
b'

Figure 1: The experiment for an oblivious transfer protocol.

$X^0 = (x_0, x_1)$, and $X^1 = (x_0, x)$ if $\mathsf{choice}(R, c, z_R) = 0$, and $X^1 = (x, x_1)$ otherwise. The experiment $\mathsf{Exp}^{\mathsf{OT}}((S, D_S), (R, D_R), \mathsf{choice}, x_0, x_1, x, z_S, z_R)$ runs as follows:

1. *Set $\mathsf{guess}_S = \mathsf{guess}_R = \mathsf{suc} = \mathsf{abort} = 0$, choose $b, c \in \{0, 1\}$ uniformly at random, and let $c^* = \mathsf{choice}(R, c, z_R)$.*

2. *Execute the oblivious transfer protocol (S, R) on input pair $((X^b, z_S), (c, z_R))$. Set $\mathsf{abort} = 1$ if some party aborts the protocol.*

3. *Run $D_S(\mathsf{view}_{S(X^b,z_S)}(R(c, z_R)))$ and $D_R(\mathsf{view}_{R(c,z_R)}(S(X^b, z_S)), X^0, X^1)$, and obtain c' and b' as output, respectively.*

4. *Set $\mathsf{guess}_S = 1$ if $c^* = c'$, and $\mathsf{guess}_R = 1$ if $b = b'$. Set $\mathsf{suc} = 1$ if either $\mathsf{out}_{R(c,z_R)}(S(X^b, z_S)) = x_{c^*}$ or $\mathsf{abort} = 1$.*

The tuple $(\mathsf{guess}_S, \mathsf{guess}_R, \mathsf{suc})$ is the outcome of the experiment.

Utility function. We assume that each party has multiple goals. The sender has the following three preferences.

- He prefers to know which of the secrets the receiver chooses to obtain.
- He does not prefer the receiver to learn both of his secrets.
- He prefers the receiver to obtain the secret that she chooses unless the protocol was aborted.

The receiver has the following preferences.

- She does not prefer the sender to know which of the secrets she chooses to obtain.
- She prefers to learn both of the sender's secrets.

- She prefers to obtain the secret that she chooses unless the protocol was aborted.

We formalize these preferences as utility functions. Note that we do not define utility functions as functions over the outcomes of the experiment; rather our utility functions are defined over the "average" outcomes of the experiment. This way of defining utility function is more natural, since parties in protocols choose their best strategy based on the average performance of the strategy, not on a single outcome of the strategy.

Definition 7 (Utility function for oblivious transfer)**.** *Let (S,R) be a two-message oblivious transfer protocol, and S' and R' algorithms.*

The utility function U_S^{OT} for the sender is a function such that $U_S^{\mathsf{OT}}(S',R) > U_S^{\mathsf{OT}}(S,R)$ if there exist probabilistic polynomial-time algorithms D_S and D_R, and $x_0,x_1,x,z_S \in \{0,1\}^*$ with $|x_0| = |x_1| = |x|$, that satisfy at least one of the following three conditions:

(S1) $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| \succ |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}|$, $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| \preceq |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}|$, *and* $\Pr[\mathsf{suc}' = 1] \succeq \Pr[\mathsf{suc} = 1]$*;*

(S2) $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| \succeq |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}|$, $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| \prec |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}|$, *and* $\Pr[\mathsf{suc}' = 1] \succeq \Pr[\mathsf{suc} = 1]$*;*

(S3) $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| \succeq |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}|$, $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| \preceq |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}|$, *and* $\Pr[\mathsf{suc}' = 1] \succ \Pr[\mathsf{suc} = 1]$,

where $(\mathsf{guess}_S, \mathsf{guess}_R, \mathsf{suc})$ *and* $(\mathsf{guess}'_S, \mathsf{guess}'_R, \mathsf{suc}')$ *are the random variables representing the outcomes of* $\mathsf{Exp}^{\mathsf{OT}}((S,D_S),(R,D_R),\mathsf{choice}_0,x_0,x_1,x,\varepsilon,\varepsilon)$ *and* $\mathsf{Exp}^{\mathsf{OT}}((S',D_S),(R,D_R),\mathsf{choice}_0,x_0,x_1,x,z_S,\varepsilon)$, *respectively, where* choice_0 *is a function that, on input* (R,c)*, outputs* c*.*

Similarly, the utility function U_R^{OT} for the receiver is a function such that $U_R^{\mathsf{OT}}(S,R') > U_R^{\mathsf{OT}}(S,R)$ if there exist probabilistic polynomial-time algorithms D_S and D_R, $x_0,x_1,x,z_R \in \{0,1\}^*$ with $|x_0| = |x_1| = |x|$, and a function $\mathsf{choice} : \{0,1\}^* \to \{0,1\}$ *that satisfy at least one of the following three conditions:*

(R1) $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| \prec |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}|$, $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| \succeq |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}|$, *and* $\Pr[\mathsf{suc} = 1] \succeq \Pr[\mathsf{suc}' = 1]$*;*

(R2) $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| \preceq |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}|$, $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| \succ |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}|$, *and* $\Pr[\mathsf{suc}' = 1] \succeq \Pr[\mathsf{suc} = 1]$*;*

(R3) $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| \preceq |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}|$, $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| \succeq |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}|$, *and* $\Pr[\mathsf{suc}' = 1] \succ \Pr[\mathsf{suc} = 1]$,

where $(\mathsf{guess}_S, \mathsf{guess}_R, \mathsf{suc})$ *and* $(\mathsf{guess}'_S, \mathsf{guess}'_R, \mathsf{suc}')$ *are the random variables representing the outcomes of* $\mathsf{Exp}^{\mathsf{OT}}((S,D_S),(R,D_R),\mathsf{choice}_0,x_0,x_1,x,\varepsilon,\varepsilon)$ *and* $\mathsf{Exp}^{\mathsf{OT}}((S,D_S),(R',D_R),\mathsf{choice},x_0,x_1,x,\varepsilon,z_R)$, *respectively.*

Note that we evaluate the success of the guess with $|\Pr[\mathsf{guess} = 1] - 1/2|$ rather than $\Pr[\mathsf{guess} = 1]$. This is because we assume that each party evaluates his own strategies as their average performance. After a single execution of the experiment, a party may prefer guess to be 0 since $\mathsf{guess} = 1$ implies the other party could successfully guess some value. However, all the values to be guessed are chosen uniformly at random, the party would prefer $\Pr[\mathsf{guess} = 1]$ to be close to $1/2$. Therefore, we use the gap between $\Pr[\mathsf{guess} = 1]$ and $1/2$ as representing the success of the guess.

Game-theoretic security. For a two-message oblivious transfer protocol OT, consider the two-party game $\Gamma^{\mathsf{OT}} = (\{S,R\},(A_S,A_R),(U_S,U_R))$ of which experiment $\mathsf{Exp}^{\mathsf{OT}}$ defined in Definition 6 where A_S is composed of all probabilistic polynomial-time algorithms, A_R is composed of all deterministic polynomial-time algorithms, and U_S and U_R are the utility functions defined in Definition 7.

We say that a protocol is game-theoretically secure if the strategy of following the protocol is in a Nash equilibrium.

Definition 8 (Game-theoretic security for oblivious transfer)**.** *A two-message oblivious transfer protocol* (S,R) *is said to be* game-theoretically secure *if* (S,R) *is in a Nash equilibrium in the game* Γ^{OT}.

3.2 Equivalence to the Cryptographic Security

We show that, for oblivious transfer protocols, the cryptographic security (Definition 2) and the game-theoretic security (Definition 8) are equivalent.

Theorem 1. *A two-message oblivious transfer protocol* OT *is cryptographically secure if and only if* OT *is game-theoretically secure.*

First, we prove that the cryptographic security implies the game-theoretic one.

Lemma 1. *If* OT *is cryptographically secure, then* OT *is game-theoretically secure.*

Proof. Assume that $\mathsf{OT} = (S,R)$ is not game-theoretically secure. Namely, (S,R) is not in a Nash equilibrium in the game Γ^{OT}. Then, there are two cases: (1) $U_S^{\mathsf{OT}}(S',R) > U_S^{\mathsf{OT}}(S,R)$ for some $S' \in A_S$; and (2) $U_R^{\mathsf{OT}}(S,R') > U_R^{\mathsf{OT}}(S,R)$ for some $R' \in A_R$.

In case (1), it follows from the definition of U_S^{OT} that either (S1), (S2), or (S3) holds. Condition (S1) implies that, by using (S',D_S) as a strategy, the sender can predict the choice bit c with probability greater than $1/2$. More specifically, $\Pr[D_S(\mathsf{view}_{S'(X^b,z_S)}(R(c))) = c] \succ 1/2$. This means that OT does not satisfy the receiver's privacy. Condition (S2) means that $|\Pr[\mathsf{guess}_R = 1] - 1/2| \succ 0$ when both parties follow the protocol. Namely, $|\Pr[D_R(\mathsf{view}_{R(c)}(S(X^b)),X^0,X^1) = b] - 1/2| \succ 0$, which implies that R breaks the sender's privacy. It follows from condition (S3) that $\Pr[\mathsf{out}_{R(c)}(S(x_0',x_1')) = x_c] \prec 1$ for some x_0',x_1'. This implies that OT does not satisfy correctness.

Next, let assume that (2) holds. Then, by the definition of U_R^{OT}, either (R1), (R2), or (R3) holds. Condition (R1) means that $|\Pr[\mathsf{guess}_S = 1] - 1/2| \succ 0$ when both parties follow the protocol. More precisely, $\Pr[D_S(\mathsf{view}_{S(X^b)}(R(c))) = c] \succ 0$, which implies that the sender can break the receiver's privacy. It follows from condition (R2) that $|\Pr[\mathsf{guess}_R' = 1] - 1/2| \succ 0$, which implies that $\Pr[D_R(\mathsf{view}_{R'(c,z_R)}(S(X^b)),X^0,X^1) = b] \succ 0$. Thus, R' breaks the sender's privacy. Condition (R3) implies that $\Pr[\mathsf{out}_{R(c)}(S(x_0',x_1')) = x_c'] \prec 1$ for some x_0',x_1'. Hence, OT does not satisfy correctness.

Thus, we have shown that if OT is not game-theoretically secure, then it is not cryptographically secure. □

Next, we show that the game-theoretic security implies the cryptographic one. Suppose that a protocol is not cryptographically secure. Then, it does not satisfy at least one of the cryptographic requirements. If only one of the properties is broken, it is not difficult to show that the protocol does not satisfy the game-theoretic security. However, when a protocol does not satisfy more than one properties in cryptographic security, a deeper consideration is needed. This is because, multiple requirements may cancel out the gain of utility, and there are possibilities that neither party gain by changing their strategies. We show that there is no such possibility in our game-theoretic security.

Lemma 2. *If* OT *is game-theoretically secure, then* OT *is cryptographically secure.*

Proof. Suppose that $\mathsf{OT} = (S,R)$ is not cryptographically secure. Let consider the following five cases.

(1) OT does not satisfy correctness.

(2) OT satisfies correctness, but does not satisfy the receiver's privacy when the sender follows S.

(3) OT satisfies correctness and the receiver's privacy when the sender follows S, but does not satisfy the receiver's privacy when the sender follows strategy $S' \neq S$.

(4) OT satisfies correctness, the receiver's privacy, but does not satisfy the sender's privacy when the receiver follows R and employs choice_0 as the choice function.

(5) OT satisfies correctness, the receiver's privacy, and the sender's privacy when the receiver follows R, but does not satisfy the sender's privacy when the receiver follows strategy $R' \neq R$.

For each case, we show that OT is not game-theoretically secure, namely, (S,R) is not in a Nash equilibrium.

In case (1), there exist $x_0, x_1 \in \{0,1\}^*$ with $|x_0| = |x_1|$ and $c \in \{0,1\}$ such that

$$\Pr[\mathsf{out}_{R(c)}(S(x_0,x_1)) = x_c] \prec 1.$$

Let D^{rand} be an algorithm that outputs a uniformly-random bit, and S^{abort} an algorithm that sends an abort message after getting a message from the receiver. Let consider the outcomes $(\mathsf{guess}_S, \mathsf{guess}_R, \mathsf{suc})$ and $(\mathsf{guess}'_S, \mathsf{guess}'_R, \mathsf{suc}')$ of the experiments $\mathsf{Exp}^{\mathsf{OT}}((S,D^{\mathsf{rand}}),(R,D^{\mathsf{rand}}),\mathsf{choice}_0,x_0,x_1,x,\varepsilon,\varepsilon)$ and $\mathsf{Exp}^{\mathsf{OT}}((S^{\mathsf{abort}},D^{\mathsf{rand}}),(R,D^{\mathsf{rand}}),\mathsf{choice}_0,x_0,x_1,x,\varepsilon,\varepsilon)$, respectively, where $x = 0^{|x_0|}$. Then, we have that

- $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| = |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}| = 0$,
- $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| = |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}| = 0$,
- $\Pr[\mathsf{suc}' = 1] = 1 \succ \Pr[\mathsf{suc} = 1]$.

By condition (S3) of U_S^{OT}, it holds that $U_S^{\mathsf{OT}}(S^{\mathsf{abort}},R) > U_S^{\mathsf{OT}}(S,R)$, which implies that (S,R) is not in a Nash equilibrium.[1]

Case (2) implies that there exist a probabilistic polynomial-time algorithm D_S, and $x_0, x_1 \in \{0,1\}^*$ with $|x_0| = |x_1|$ such that

$$\Pr[D_S(\mathsf{view}_{S(x_0,x_1)}(R(c))) = c] \succ \frac{1}{2},$$

where $c \in \{0,1\}$ is chosen uniformly at random. Let R^{abort} be an algorithm that sends an abort message before sending the first message. Let consider the outcomes $(\mathsf{guess}_S, \mathsf{guess}_R, \mathsf{suc})$ and $(\mathsf{guess}'_S, \mathsf{guess}'_R, \mathsf{suc}')$ of the experiments $\mathsf{Exp}^{\mathsf{OT}}((S,D_S),(R,D^{\mathsf{rand}}),\mathsf{choice}_0,x_0,x_1,x,\varepsilon,\varepsilon)$ and $\mathsf{Exp}^{\mathsf{OT}}((S,D_S),(R^{\mathsf{abort}},D^{\mathsf{rand}}),\mathsf{choice}_0,x_0,x_1,x,\varepsilon,\varepsilon)$, respectively, where $x = 0^{|x_0|}$. Then, we have that

- $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| = 0 \prec |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}|$,
- $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| = |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}| = 0$,
- $\Pr[\mathsf{suc}' = 1] = 1 \approx \Pr[\mathsf{suc} = 1]$.

[1] We can also show that (S,R) is not in a Nash equilibrium based on condition (R3) of U_R^{OT} by using almost the same argument as above.

By condition (R1) of U_R^{OT}, it holds that $U_R^{\mathsf{OT}}(S, R^{\mathsf{abort}}) > U_R^{\mathsf{OT}}(S, R)$, which implies that (S, R) is not in a Nash equilibrium.

In case (3), the receiver's privacy holds for a semi-honest sender, but not for a malicious sender. Specifically, there exist probabilistic polynomial-time algorithms S' and D_S, and $x_0, x_1, z_S \in \{0,1\}^*$ with $|x_0| = |x_1|$ such that

$$\Pr[D_S(\mathsf{view}_{S'(x_0,x_1,z_S)}(R(c))) = c] \succ \frac{1}{2},$$

where $c \in \{0,1\}$ is chosen uniformly at random. Let S'' be an algorithm that simulates S', and sends an abort message right before sending his message. Consider the experiments $\mathsf{Exp}^{\mathsf{OT}}((S, D_S), (R, D^{\mathsf{rand}}), \mathsf{choice}_0, x_0, x_1, x, z_S, \varepsilon)$ and $\mathsf{Exp}^{\mathsf{OT}}((S'', D_S), (R, D^{\mathsf{rand}}), \mathsf{choice}_0, x_0, x_1, x, z_S, \varepsilon)$, where $x = 0^{|x_0|}$, and their corresponding outcomes $(\mathsf{guess}_S, \mathsf{guess}_R, \mathsf{suc})$ and $(\mathsf{guess}'_S, \mathsf{guess}'_R, \mathsf{suc}')$. It holds that

- $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| \succ |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}| \approx 0,$
- $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| = |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}| = 0,$
- $\Pr[\mathsf{suc}' = 1] = 1 \approx \Pr[\mathsf{suc} = 1].$

It follows from condition (S1) of U_S^{OT} that $U_S^{\mathsf{OT}}(S'', R) > U_S^{\mathsf{OT}}(S, R)$. Hence, (S, R) is not in a Nash equilibrium.

Next, we consider case (4). In this case, there exist a probabilistic polynomial-time algorithm D_R and $x_0, x_1, x \in \{0,1\}^*$ with $|x_0| = |x_1| = |x|$ such that

$$\Pr[D_R(\mathsf{view}_{R(c)}(S(X^b)), X^0, X^1) = b] \succ \frac{1}{2},$$

where $X^0 = (x_0, x_1)$, $X^1 = (x_0, x)$ if $c = 0$, and $X^1 = (x, x_1)$ otherwise, and $b, c \in \{0,1\}$ are chosen uniformly at random. Let consider the experiments $\mathsf{Exp}^{\mathsf{OT}}((S, D^{\mathsf{rand}}), (R, D_R), \mathsf{choice}_0, x_0, x_1, x, \varepsilon, \varepsilon)$ and $\mathsf{Exp}^{\mathsf{OT}}((S^{\mathsf{abort}}, D^{\mathsf{rand}}), (R, D_R), \mathsf{choice}_0, x_0, x_1, x, \varepsilon, \varepsilon)$, and their corresponding outcomes $(\mathsf{guess}_S, \mathsf{guess}_R, \mathsf{suc})$ and $(\mathsf{guess}'_S, \mathsf{guess}'_R, \mathsf{suc}')$. Then, it holds that

- $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| = |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}| = 0,$
- $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| = 0 \prec |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}|,$
- $\Pr[\mathsf{suc}' = 1] = 1 \approx \Pr[\mathsf{suc} = 1].$

By condition (S2) of U_S^{OT}, we have that $U_S^{\mathsf{OT}}(S^{\mathsf{abort}}, R) > U_S^{\mathsf{OT}}(S, R)$. Therefore, (S, R) is not in a Nash equilibrium.

Finally, let consider case (5). There exist a deterministic polynomial-time algorithm R' and probabilistic polynomial-time algorithm D_R, and $x_0, x_1, x, z_R \in \{0,1\}^*$ with $|x_0| = |x_1| = |x|$ such that for any function $\mathsf{choice} : \{0,1\}^* \to \{0,1\}$, it holds that

$$\Pr[D_R(\mathsf{view}_{R'(z_R)}(S(X^b)), X^0, X^1) = b] \succ \frac{1}{2},$$

where $X^0 = (x_0, x_1)$, $X^1 = (x_0, x)$ if $c^* = 0$, and $X^1 = (x, x_1)$ otherwise, $c^* = \mathsf{choice}(R', c, z_R)$, and $b, c \in \{0,1\}$ are chosen uniformly at random. Consider an algorithm R'' that simulates R' and sends an abort message after receiving a message from the sender. Let $(\mathsf{guess}_S, \mathsf{guess}_R, \mathsf{suc})$ and $(\mathsf{guess}'_S, \mathsf{guess}'_R, \mathsf{suc}')$ be the outcomes of the experiments $\mathsf{Exp}^{\mathsf{OT}}((S, D^{\mathsf{rand}}), (R, D_R), \mathsf{choice}_0, x_0, x_1, x, \varepsilon, \varepsilon)$ and $\mathsf{Exp}^{\mathsf{OT}}((S^{\mathsf{abort}}, D^{\mathsf{rand}}), (R'', D_R), \mathsf{choice}, x_0, x_1, x, \varepsilon, z_R)$, respectively. Then, we have that

- $|\Pr[\mathsf{guess}'_S = 1] - \frac{1}{2}| = |\Pr[\mathsf{guess}_S = 1] - \frac{1}{2}| = 0$,
- $|\Pr[\mathsf{guess}'_R = 1] - \frac{1}{2}| \succ 0 \approx |\Pr[\mathsf{guess}_R = 1] - \frac{1}{2}|$,
- $\Pr[\mathsf{suc}' = 1] = 1 \approx \Pr[\mathsf{suc} = 1]$.

It follows from condition (R2) of U_R^{OT} that $U_R^{\mathsf{OT}}(S,R'') > U_R^{\mathsf{OT}}(S,R)$. Thus, (S,R) is not in a Nash equilibrium.

In every case, we have shown that (S,R) is not in a Nash equilibrium. Therefore, the statement follows.

□

3.3 Discussion

Unnecessary conditions in utility functions. In the proof of Lemma 2, we did not use the condition (R3) of U_R^{OT}. It is not difficult to see that Lemma 1 holds even if condition (R3) is not included in U_R^{OT}. This implies that the equivalence holds even if condition (R3) is excluded from U_R^{OT}. The proof of Lemma 2 can be completed by using condition (S3) of U_S^{OT} instead of (R3) of U_R^{OT}. Thus, we can also say that the equivalence holds if condition (S3) is excluded from U_S^{OT}, where (R3) should be included in U_R^{OT}. In other words, the equivalence holds between the game-theoretic security and cryptographic one as long as at least one of the parties prefers the receiver to obtain the secret that she chose.

Note, however, that if some party has the opposite preference to, say, condition (S3), the equivalence does not hold. If the sender prefers the receiver to obtain the secret which was not chosen by the receiver, then Lemma 1 does not hold while Lemma 2 holds. In this case, a cryptographically-secure protocol does not achieve a Nash equilibrium because the receiver can obtain the secret she chose, which is not preferred by the sender. Conversely, if a given protocol satisfies the game-theoretic security for the sender with this utility, it implies that the protocol achieves a Nash equilibrium for the receiver who has a preference for correctness. Therefore, it satisfies the cryptographic security.

Abort after completing the protocol. In case (5) of the proof of Lemma 2, we use the fact that the receiver can abort the protocol even after receiving the second message from the sender. Thus, if such an abort is not allowed, the equivalence may not hold.

4 Game-Theoretic Security for Commitment

In this section, we provide a game-theoretic characterization of commitment protocols. Our game-theoretic security is shown to be equivalent to the cryptographic security. We also discuss the implication of our equivalence result.

4.1 Definition

First, we define an experiment for the execution of a commitment protocol. Then, we consider natural preferences of the sender and the receiver. By using a Nash equilibrium as a solution concept, we define the game-theoretic security of commitment.

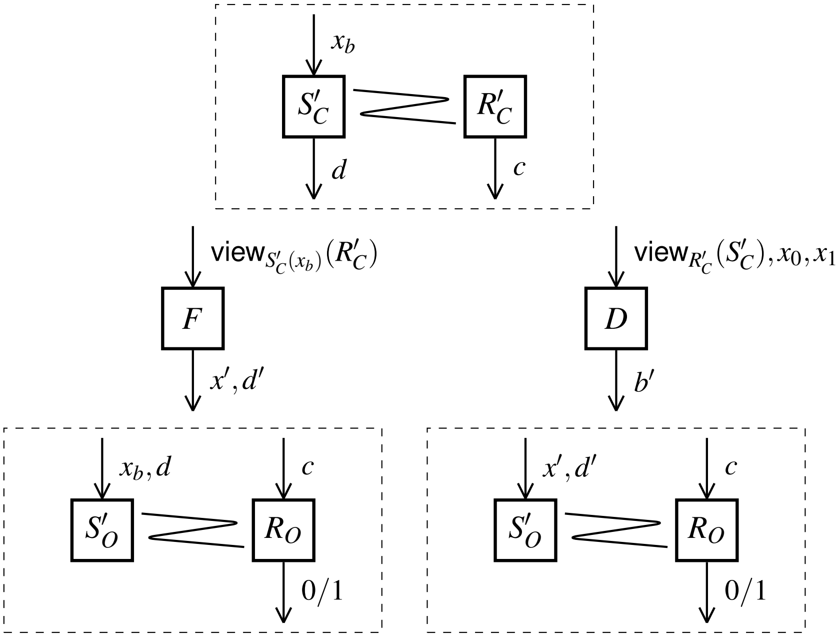

Figure 2: The experiment for a commitment protocol.

Experiment. Let $\mathsf{Com} = ((S_C, S_O), (R_C, R_O))$ be a commitment protocol. We define an experiment between a sender and a receiver. Both the sender and the receiver have three algorithms (S_C, S_O, F) and (R_C, R_O, D), respectively. These algorithms interact as follows.

First, the sender and the receiver execute a commit phase by using S_C and R_C, where S_C is given input $x_b \in \{0,1\}^t$, where $x_0, x_1 \in \{0,1\}^t$ are possible inputs, and $b \in \{0,1\}$ is chosen uniformly at random. Let c and d be the commitment and decommitment strings generated in this phase. Then, a distinguisher D of the receiver tries to guess the committed string x_b based on the view of R_C in the commit phase and possible inputs x_0, x_1. After that, a decommitment finder F of the sender tries to generate (x', d') so that $x' (\neq x_b)$ can be opened by using d' as the decommitment string. Then, the open phase is executed twice, where the first one checks if x_b can be opened with the decommitment string d, and the second one does if x' with d'.

We formally define the experiment for commitment protocols. (See also Figure 2.)

Definition 9 (Experiment for commitment)**.** *Let S_C, S_O, F, R_C, R_O, and D be algorithms, $x_0, x_1 \in \{0,1\}^t$, and $z_S, z_R \in \{0,1\}^*$. The experiment $\mathsf{Exp}^{\mathsf{Com}}((S_C, S_O, F), (R_C, R_O, D), x_0, x_1, z_S, z_R)$ is executed as follows.*

1. *Set $\mathsf{guess} = \mathsf{amb} = \mathsf{suc} = \mathsf{abort}_C = \mathsf{abort}_O = 0$, and choose $b \in \{0,1\}$ uniformly at random.*
2. *Execute a commit phase by using $S_C(x_b, z_S)$ and $R_C(z_R)$. Let c and d be the commitment and decommitment strings, respectively, that are generated during the execution. Set $\mathsf{abort}_C = 1$ if some party aborts the protocol.*
3. *If $\mathsf{abort}_C = 0$, run $D(\mathsf{view}_{R_C(z_R)}(S_C(x_b, z_S)), x_0, x_1)$ and $F(\mathsf{view}_{S_C(x_b, z_S)}(R_C(z_R)))$, and obtain as output b' and (x', d'), respectively. Otherwise, choose $b' \in \{0,1\}$ uniformly at random.*
4. *If $\mathsf{abort}_C = 0$, execute an open phase twice, where the first one is done between $S_O(x_b, d, z_S)$ and $R_O(c)$, and the second one is between $S_O(x', d', z_S)$ and $R_O(c)$. Let o and o' be the outputs of R_O in the*

first and second executions, respectively. If some party aborts in the first (and second) interaction(s), set $\mathsf{abort}_O = 1$ *and* $o = 0$ *(and* $o' = 0$*).*

If $\mathsf{abort}_C = 1$*, set* $o = o' = 0$.

5. *Set* $\mathsf{amb} = 1$ *if* $x_b \neq x'$ *and* $o = o' = 1$. *Set* $\mathsf{suc} = 1$ *if either* $o = 1$, $\mathsf{abort}_C = 1$, *or* $\mathsf{abort}_O = 1$. *Set* $\mathsf{guess} = 1$ *if* $b = b'$.

The tuple $(\mathsf{guess}, \mathsf{amb}, \mathsf{suc})$ *is the outcome of this experiment.*

Utility functions. We assume that each party of commitment has multiple goals. The sender has the following two preferences:

- He does not prefer the receiver to know the committed string x_b before executing the open phase.
- On executing the open phase, he prefers to be able to choose a string to be opened.

The receiver has the following three preferences:

- She prefers to learn the committed string x_b before executing the open phase.
- She does not prefer the sender to be able to choose a string to be opened in the open phase.
- She prefers to open the true committed string x_b unless the protocol was aborted.

We formalize these preferences as utility functions. As in the case of oblivious transfer protocols, the utility functions are defined over the average outcomes of the experiments.

Definition 10 (Utility functions for commitment)**.** *Let* $((S_C, S_O), (R_C, R_O))$ *be a commitment protocol, and* S'_C, S'_O, R'_C, R'_O *algorithms. The utility function* U_S^{Com} *for the sender is a function such that* $U_S^{\mathsf{Com}}((S'_C, S'_O), (R_C, R_O)) > U_S^{\mathsf{Com}}((S_C, S_O), (R_C, R_O))$ *if there exist probabilistic polynomial-time algorithms F and D,* $x_0, x_1 \in \{0,1\}^t$*, and* $z_S \in \{0,1\}^*$*, that satisfy at least one of the following two conditions:*

(S1) $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| \prec \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right|$ *and* $\Pr[\mathsf{amb}' = 1] \succeq \Pr[\mathsf{amb} = 1]$*;*

(S2) $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| \preceq \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right|$ *and* $\Pr[\mathsf{amb}' = 1] \succ \Pr[\mathsf{amb} = 1]$,

where $(\mathsf{guess}, \mathsf{amb}, \mathsf{suc})$ *and* $(\mathsf{guess}', \mathsf{amb}', \mathsf{suc}')$ *are the random variables representing the outcomes of* $\mathsf{Exp}^{\mathsf{Com}}((S_C, S_O, F), (R_C, R_O, D), x_0, x_1, \varepsilon, \varepsilon)$ *and* $\mathsf{Exp}^{\mathsf{Com}}((S'_C, S'_O, F), (R_C, R_O, D), x_0, x_1, z_S, \varepsilon)$*, respectively.*

The utility function U_R^{Com} *for the receiver is a function such that* $U_R^{\mathsf{Com}}((S_C, S_O), (R'_C, R'_O)) > U_R^{\mathsf{Com}}((S_C, S_O), (R_C, R_O))$ *if there exist probabilistic polynomial-time algorithms F and D,* $x_0, x_1 \in \{0,1\}^t$*, and* $z_R \in \{0,1\}^*$*, that satisfy at least one the following three conditions:*

(R1) $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| \succ \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right|$, $\Pr[\mathsf{amb}' = 1] \preceq \Pr[\mathsf{amb} = 1]$, *and* $\Pr[\mathsf{suc}' = 1] \succeq \Pr[\mathsf{suc} = 1]$*;*

(R2) $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| \succeq \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right|$, $\Pr[\mathsf{amb}' = 1] \prec \Pr[\mathsf{amb} = 1]$, *and* $\Pr[\mathsf{suc}' = 1] \succeq \Pr[\mathsf{suc} = 1]$*;*

(R3) $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| \succeq \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right|$, $\Pr[\mathsf{amb}' = 1] \preceq \Pr[\mathsf{amb} = 1]$, *and* $\Pr[\mathsf{suc}' = 1] \succ \Pr[\mathsf{suc} = 1]$,

where $(\mathsf{guess}, \mathsf{amb}, \mathsf{suc})$ *and* $(\mathsf{guess}', \mathsf{amb}', \mathsf{suc}')$ *are the random variables representing the outcomes of* $\mathsf{Exp}^{\mathsf{Com}}((S_C, S_O, F), (R_C, R_O, D), x_0, x_1, \varepsilon, \varepsilon)$ *and* $\mathsf{Exp}^{\mathsf{Com}}((S_C, S_O, F), (R'_C, R'_O, D), x_0, x_1, \varepsilon, z_R)$*, respectively.*

Game-theoretic security. For a commitment protocol Com, let define the two-party game $\Gamma^{\mathsf{Com}} = (\{S,R\},(A_S,A_R),(U_S,U_R))$ in which the experiment $\mathsf{Exp}^{\mathsf{Com}}$ defined in Definition 9 is executed, both A_S and A_R are composed of pairs of all probabilistic polynomial-time algorithms, and U_S and U_R are the utility functions defined in Definition 10.

We say that a protocol is game-theoretically secure if the strategy of following the protocol description is in a Nash equilibrium.

Definition 11 (Game-theoretic security for commitment)**.** *A commitment protocol* $((S_C,S_O),(R_C,R_O))$ *is said to be* game-theoretically secure *if* $((S_C,S_O),(R_C,R_O))$ *is in a Nash equilibrium in the game* Γ^{Com}.

4.2 Equivalence to the Cryptograhic Security

In this section, we prove the equivalence between the cryptographic security (Definition 4) and the game-theoretic security (Definition 11) for commitment protocols.

Theorem 2. *A commitment protocol* Com *is cryptographically secure if and only if* Com *is game-theoretically secure.*

First, we show that the cryptographic security implies the game-theoretic security.

Lemma 3. *If* Com *is cryptographically secure, then* Com *is game-theoretically secure.*

Proof. Let assume that $\mathsf{Com} = ((S_C,S_O),(R_C,R_O))$ is not game-theoretically secure, namely, $((S_C,S_O),(R_C,R_O))$ is not in a Nash equilibrium. Then, there are two cases: (1) $U_S^{\mathsf{Com}}((S'_C,S'_O),(R_C,R_O)) > U_S^{\mathsf{Com}}((S_C,S_O),(R_C,R_O))$ for some $(S'_C,S'_O) \in A_S$; and (2) $U_R^{\mathsf{Com}}((S_C,S_O),(R'_C,R'_O)) > U_S^{\mathsf{Com}}((S_C,S_O),(R_C,R_O))$ for some $(R'_C,R'_O) \in A_R$.

In case (1), it follows from the definition of U_S^{Com} that either (S1) or (S2) holds. We observe that condition (S1) implies that $|\Pr[\mathsf{guess} = 1] - 1/2| \succ 0$. Then, it holds that $|\Pr[D(\mathsf{view}_{R_C}(S_C(x_b)),x_0,x_1) = b] - 1/2| \succ 0$, where $b \in \{0,1\}$ is chosen uniformly at random. This means that Com does not satisfy hiding property. Condition (S2) implies that $\Pr[\mathsf{amb}' = 1] \succ 0$, which means that $\Pr[\mathsf{out}_{R_O(c)}(S'_O(x_b,d,z_S)) = \mathsf{out}_{R_O(c)}(S'_O(x',d',z_S)) = 1] \succ 0$, where c and d are the commitment and decommitment string generated by the interaction between $S'_C(x_b,z_S)$ and R_C, (x',d') is the output of $F(\mathsf{view}_{S'_C(x_b,z_S)}(R_C))$, and $b \in \{0,1\}$ is chosen uniformly at random. This implies that Com does not satisfy binding property.

Next, we consider case (2). It follows from the definition of U_R^{Com} that either (R1), (R2), or (R3) holds. Condition (R1) implies that $|\Pr[\mathsf{guess}' = 1] - 1/2| \succ 0$. This means that $\Pr[D(\mathsf{view}_{R'_C(z_R)}(S_C(x_b,z_S)),x_0,x_1) = b] \succ 1/2$, where $b \in \{0,1\}$ is chosen uniformly at random. Hence, Com does not satisfy hiding property. Condition (R2) implies that $\Pr[\mathsf{amb} = 1] \succ 0$. This means that $\Pr[\mathsf{out}_{R_O(c)}(S_O(x_b,d)) = \mathsf{out}_{R_O(c)}(S_O(x',d')) = 1] \succ 0$, where c and d are the commitment and decommitment strings generated by the interaction between $S_C(x_b)$ and R_C, (x',d') is the output of $F(\mathsf{view}_{S_C(x_b)}(R_C))$ with $x' \neq x_b$, and $b \in \{0,1\}$ is chosen uniformly at random. This implies that Com does not satisfy binding property. Finally, let consider condition (R3), which implies $\Pr[\mathsf{suc} = 1] \prec 1$. Then, we have that $\Pr[\mathsf{out}_{R_O(c)}(S_O(x_b,d)) = 1] \prec 1$, where c and d are the commitment and decommitment strings generated by $S_C(x_b)$ and R_C, and $b \in \{0,1\}$ is chosen uniformly at random. This means that Com does not satisfy correctness property.

In every case, we have shown that Com is not cryptographically secure. Therefore, the statement follows. □

Next, we show that the game-theoretic security implies the cryptographic security.

Lemma 4. *If* Com *is game-theoretically secure, then* Com *is cryptographically secure.*

Proof. Suppose that $\mathsf{Com} = ((S_C, S_O), (R_C, R_O))$ is not cryptographically secure. We consider the following five cases, and show that Com is not game-theoretically secure in each case.

(1) Com does not satisfy correctness.

(2) Com satisfies correctness, but does not satisfy binding property for (S_C, S_O).

(3) Com satisfies correctness and binding property for (S_C, S_O), but does not satisfy binding property for some $(S'_C, S'_O) \neq (S_C, S_O)$.

(4) Com satisfies correctness and binding property, but does not satisfy hiding property for (R_C, R_O).

(5) Com satisfies correctness, binding property, and hiding property for (R_C, R_O), but does not satisfy hiding property for some $(R'_C, R'_O) \neq (R_C, R_O)$.

In case (1), for some $x \in \{0,1\}^t$, it holds that

$$\Pr[\mathsf{out}_{R_O(c)}(S_O(x,d)) = 1] \prec 1,$$

where c and d are the commitment and decommitment strings generated during the interaction between $S_C(x)$ and R_C. Let D^{rand} be an algorithm that outputs $b \in \{0,1\}$ uniformly at random, F_0 an algorithm that, on input the view including (x,c,d), outputs (x,d), and R_C^{abort} a strategy of sending an abort message right after starting the commit phase. We denote by $(\mathsf{guess}, \mathsf{amb}, \mathsf{suc})$ and $(\mathsf{guess}', \mathsf{amb}', \mathsf{suc}')$ the outcomes of the experiments $\mathsf{Exp}^{\mathsf{Com}}((S_C, S_O, F_0), (R_C, R_O, D^{\mathsf{rand}}), x, x, \varepsilon, \varepsilon)$ and $\mathsf{Exp}^{\mathsf{Com}}((S_C, S_O, F_0), (R_C^{\mathsf{abort}}, R_O, D^{\mathsf{rand}}), x, x, \varepsilon, \varepsilon)$, respectively. Note that, when the receiver follows R_C^{abort}, $\mathsf{abort}_C = 1$ in the experiment, and thus the tests for F and D will not be checked. Thus, we have that

- $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| = \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right| = 0$,
- $\Pr[\mathsf{amb}' = 1] = 0 \preceq \Pr[\mathsf{amb} = 1]$,
- $\Pr[\mathsf{suc}' = 1] = 1 \succ \Pr[\mathsf{suc} = 1]$.

By condition (R3) of U_R^{Com}, we have that $U_R^{\mathsf{Com}}((S_C, S_O), (R_C^{\mathsf{abort}}, R_O)) > U_R^{\mathsf{Com}}((S_C, S_O), (R_C, R_O))$, which implies that the tuple $((S_C, S_O), (R_C, R_O))$ is not in a Nash equilibrium.

Next, we consider case (2). In this case, there is a probabilistic polynomial-time decommitment finder F and $x \in \{0,1\}^t$ so that

$$\Pr[\mathsf{out}_{R_O(c)}(S_O(x_0,d_0)) = \mathsf{out}_{R_O(c)}(S_O(x_1,d_1)) = 1] \succ 0,$$

where c and d are the commitment and decommitment strings generated by $S_C(x)$ and R_C, (x', d') is the output of $F(\mathsf{view}_{S_C(x)}(R_C))$. Let $(\mathsf{guess}, \mathsf{amb}, \mathsf{suc})$ and $(\mathsf{guess}', \mathsf{amb}', \mathsf{suc}')$ be the outcomes of the experiments $\mathsf{Exp}^{\mathsf{Com}}((S_C, S_O, F), (R_C, R_O, D^{\mathsf{rand}}), x, x', \varepsilon, \varepsilon)$ and $\mathsf{Exp}^{\mathsf{Com}}((S_C, S_O, F), (R_C^{\mathsf{abort}}, R_O, D^{\mathsf{rand}}), x, x', \varepsilon, \varepsilon)$, respectively, where $x' \in \{0,1\}^t \setminus \{x\}$ is arbitrary. Then, we have that

- $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| = \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right| = 0$,
- $\Pr[\mathsf{amb}' = 1] = 0 \prec \Pr[\mathsf{amb} = 1]$,

- $\Pr[\mathsf{suc}'=1]=1\approx\Pr[\mathsf{suc}=1]$.

Hence, by condition (R2) of U_R^{Com}, it holds that $U_R^{\mathsf{Com}}((S_C,S_O),(R^{\mathsf{abort}},R_O))>U_R^{\mathsf{Com}}((S_C,S_O),(R_C,R_O))$. Therefore, the tuple $((S_C,S_O),(R_C,R_O))$ is not in a Nash equilibrium.

In case (3), there exist a probabilistic polynomial-time algorithm F, $x\in\{0,1\}^t$, and $z\in\{0,1\}^*$, it holds that

$$\Pr[\mathsf{out}_{R_O(c)}(S_O'(x,d,z))=\mathsf{out}_{R_O(c)}(S_O'(x',d',z))=1]\succ 0,$$

where c and d are the commitment and decommitment strings generated by $S_C'(x,z)$ and R_C, and (x',d') is the output of $F(\mathsf{view}_{S_C'(x,z)}(R_C))$ with $x'\neq x$. Let $(\mathsf{guess},\mathsf{amb},\mathsf{suc})$ and $(\mathsf{guess}',\mathsf{amb}',\mathsf{suc}')$ be the outcomes of the experiments $\mathsf{Exp}^{\mathsf{Com}}((S_C,S_O,F),(R_C,R_O,D^{\mathsf{rand}}),x,\varepsilon,\varepsilon)$ and $\mathsf{Exp}^{\mathsf{Com}}((S_C',S_O',F),(R_C,R_O,D^{\mathsf{rand}}),x,z,\varepsilon)$, respectively. We have that

- $\left|\Pr[\mathsf{guess}'=1]-\frac{1}{2}\right|=\left|\Pr[\mathsf{guess}=1]-\frac{1}{2}\right|=0$,
- $\Pr[\mathsf{amb}'=1]\succ\Pr[\mathsf{amb}=1]\approx 0$.

By condition (S2) of U_S^{Com}, it holds that

$$U_S^{\mathsf{Com}}((S_C',S_O'),(R_C,R_O))>U_S^{\mathsf{Com}}((S_C,S_O),(R_C,R_O)).$$

Thus, the tuple $((S_C,S_O),(R_C,R_O))$ is not in a Nash equilibrium.

We consider case (4), in which the receiver can break hiding property with the honest strategy. Then, there is a probabilistic polynomial-time algorithm D and $x_0,x_1\in\{0,1\}^t$ so that

$$\left|\Pr[D(\mathsf{view}_{R_C}(S_C(x_b),x_0,x_1))=b]-\frac{1}{2}\right|\succ 0,$$

where $b\in\{0,1\}$ is chosen uniformly at random. Let S_C^{abort} be the strategy of sending an abort message right after starting the commit phase. We denote by $(\mathsf{guess},\mathsf{amb},\mathsf{suc})$ and $(\mathsf{guess}',\mathsf{amb}',\mathsf{suc}')$ the outcomes of the experiments $\mathsf{Exp}^{\mathsf{Com}}((S_C,S_O,F_0),(R_C,R_O,D),x_0,x_1,\varepsilon,\varepsilon)$ and $\mathsf{Exp}^{\mathsf{Com}}((S_C^{\mathsf{abort}},S_O,F_0),(R_C,R_O,D),x_0,x_1,\varepsilon,\varepsilon)$, respectively. Then, we have that

- $\left|\Pr[\mathsf{guess}'=1]-\frac{1}{2}\right|=0\prec\left|\Pr[\mathsf{guess}=1]-\frac{1}{2}\right|$,
- $\Pr[\mathsf{amb}'=1]=0\approx\Pr[\mathsf{amb}=1]$.

Hence, by condition (S1) of U_S^{Com}, it holds that $U_S^{\mathsf{Com}}((S_C^{\mathsf{abort}},S_O),(R_C,R_O))>U_S^{\mathsf{Com}}((S_C,S_O),(R_C,R_O))$, which implies that the tuple $((S_C,S_O),(R_C,R_O))$ is not in a Nash equilibrium.

In case (5), there exist probabilistic polynomial-time algorithms $R_C'(\neq R_C)$ and D, $x_0,x_1\in\{0,1\}^t$, and $z\in\{0,1\}^*$, such that

$$\left|\Pr[D(\mathsf{view}_{R_C'(z)}(S_C(x_b)),x_0,x_1)=b]-\frac{1}{2}\right|\succ 0,$$

and

$$\left|\Pr[D(\mathsf{view}_{R_C}(S_C(x_b)),x_0,x_1)=b]-\frac{1}{2}\right|\approx 0,$$

where $b\in\{0,1\}$ is chosen uniformly at random. Let R_O^{abort} be the strategy of sending an abort message right after starting the open phase. We denote by $(\mathsf{guess},\mathsf{amb},\mathsf{suc})$ and $(\mathsf{guess}',\mathsf{amb}',\mathsf{suc}')$ the outcomes of the games $\mathsf{Exp}^{\mathsf{Com}}((S_C,S_O,F_0),(R_C,R_O,D),x_0,x_1,\varepsilon,\varepsilon)$ and $\mathsf{Exp}^{\mathsf{Com}}((S_C,S_O,F_0),(R_C',R_O^{\mathsf{abort}},D),x_0,x_1,\varepsilon,z)$, respectively. Then, it holds that

- $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| \succ 0 \approx \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right|$,
- $\Pr[\mathsf{amb}' = 1] = 0 \approx \Pr[\mathsf{amb} = 1]$,
- $\Pr[\mathsf{suc}' = 1] = 1 \approx \Pr[\mathsf{suc} = 1]$.

It follows from condition (R1) of U_R^{Com} that $U_R^{\mathsf{Com}}((S_C, S_O), (R'_C, R_O^{\mathsf{abort}})) > U_R^{\mathsf{Com}}((S_C, S_O), (R_C, R_O))$, which implies that the tuple $((S_C, S_O), (R_C, R_O))$ is not in a Nash equilibrium.

In every case, we have shown that the tuple $((S_C, S_O), (R_C, R_O))$ is not in a Nash equilibrium. Thus, the statement follows. □

4.3 Discussion

Sender's preference for correctness. In our characterization, only the receiver has the preference corresponding to correctness. Let us consider the case in which the sender has the following preference.

- The sender prefers the receiver to open the true committed string x_b in the open phase unless the protocol was aborted.

Then, U_S^{Com} will be changed so that $U_S^{\mathsf{Com}}((S'_C, S'_O), (R_C, R_O)) > U_S^{\mathsf{Com}}((S_C, S_O), (R_C, R_O))$ holds if there exist probabilistic polynomial-time algorithms F and D, $x_0, x_1 \in \{0,1\}^t$, and $z_S \in \{0,1\}^*$, that satisfy at least one of the following three conditions:

(S1') $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| \prec \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right|$, $\Pr[\mathsf{amb}' = 1] \succeq \Pr[\mathsf{amb} = 1]$, and $\Pr[\mathsf{suc}' = 1] \succeq \Pr[\mathsf{suc} = 1]$;

(S2') $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| \preceq \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right|$, $\Pr[\mathsf{amb}' = 1] \succ \Pr[\mathsf{amb} = 1]$, and $\Pr[\mathsf{suc}' = 1] \succeq \Pr[\mathsf{suc} = 1]$;

(S3') $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| \preceq \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right|$, $\Pr[\mathsf{amb}' = 1] \succeq \Pr[\mathsf{amb} = 1]$, and $\Pr[\mathsf{suc}' = 1] \succ \Pr[\mathsf{suc} = 1]$;

where $(\mathsf{guess}, \mathsf{amb}, \mathsf{suc})$ and $(\mathsf{guess}', \mathsf{amb}', \mathsf{suc}')$ are the random variables representing the outcomes of $\mathsf{Exp}^{\mathsf{Com}}((S_C, S_O, F), (R_C, R_O, D), x_0, x_1, \varepsilon, \varepsilon)$ and $\mathsf{Exp}^{\mathsf{Com}}((S'_C, S'_O, F), (R_C, R_O, D), x_0, x_1, z_S, \varepsilon)$, respectively.

Under the above conditions on U_S^{Com}, we cannot prove the equivalence to the cryptographic security. More precisely, we can show Lemma 3 in a similar way as the above proof. However, we cannot prove Lemma 4 under the new utility function. Specifically, in case (3) in the proof of Lemma 4, it is assumed that there exists (S_C^*, S_O^*) that breaks binding property. We need to choose (S'_C, S'_O) so that

- $\left|\Pr[\mathsf{guess}' = 1] - \frac{1}{2}\right| \approx \left|\Pr[\mathsf{guess} = 1] - \frac{1}{2}\right|$,
- $\Pr[\mathsf{amb}' = 1] \succ \Pr[\mathsf{amb} = 1] \approx 0$,
- $\Pr[\mathsf{suc}' = 1] \succeq \Pr[\mathsf{suc} = 1]$.

The first condition can be easily satisfied by using D^{rand} as a part of the receiver's strategy. To satisfy the second condition, we need to use (S_C^*, S_O^*). The problem is how to achieve the last condition. Suppose that the protocol has the property such that whenever the sender breaks binding property, the correctness is not preserved. It implies that the last condition cannot be satisfied when the second condition holds. Thus, the lemma does not hold under the new utility function.

Furthermore, let consider the sender who has the opposite preference regarding correctness. Namely, the sender does not prefer the receiver to obtain the committed string. In this case, Lemma 3 does not hold, while Lemma 4 holds. Since a cryptographically-secure protocol satisfies the correctness property, the rational sender does not prefer to following the protocol, which implies Lemma 3 does not hold. Conversely, if a given protocol achieve a Nash equilibrium even if the sender has the opposite preference for correctness, since the receiver has the preference for correctness, the protocol satisfies the cryptographic security.

The above examples illustrate the flexibility and generality of game-theoretic security. The party having the utility U_S^{Com} defined with conditions (S1'), (S2'), and (S3') can be considered a party who does not prefer to breaking binding without preserving correctness. The sender who has the opposite preference to correctness is a party who prefer to breaking binding and correctness simultaneously. By considering other utility functions, we could define various levels of security for commitment protocols.

5 Conclusion and Future Work

This paper has studied oblivious transfer and commitment using game-theoretic concepts. Based on the previous work by Asharov et al. [2], we have extended the game-theoretic characterization of cryptographic protocols. In our game-theoretic security, the parties can consider the trade-off among the preferences. Our conceptual contribution includes capturing the computational security by plain Nash equilibria. We have shown the equivalence between our security and the standard cryptographic one both for oblivious transfer and commitment. The equivalence implies that our formalization is a novel way to capture the standard security in terms of game theory. To put it another way, the equivalence claims that the standard security is reasonable even if we consider rational parties.

Our results illustrate the generality of game-theoretic formalizations. The game-theoretic security can be easily strengthened and weakened by considering various solution concepts and utility functions. For example, we can define a stronger security by employing a subgame perfect equilibrium, which is a preferable solution concept than a Nash equilibrium in extensive-form games. Several solution concepts are studied for dealing with computationally bounded strategies [11, 20]. Exploring the possibilities of applying various solution concepts in our formalization is an interesting future work. As discussed in Section 4.3, our game-theoretic security can also become weaker by considering other utility functions. Another future work includes investigating meaningful weaker security for cryptographic protocols that utilize rational parties' consideration on trade-offs among multiple goals.

Finally, we note that although oblivious transfer and commitment are fundamental protocols in cryptographic protocols, they are rarely considered the final protocol, and are mostly used as building blocks of other protocols. Since a game-theoretic consideration of protocol participants is most useful in the final protocol, it is necessary to investigate the game-theoretic security for more advanced protocols.

Acknowledgments

This work was supported in part by JSPS/MEXT Grant-in-Aid for Scientific Research Numbers 23500010, 23700010, 24240001, 25106509, 15H00851, and 16H01705; JST, CREST, Mathematical Modelling for Next-Generation Cryptography; and ASPIRE League Research Grant.

References

[1] I. Abraham, D. Dolev, and J. Y. Halpern. Distributed protocols for leader election: A game-theoretic perspective. In Y. Afek, editor, *Distributed Computing - 27th International Symposium, DISC 2013, Jerusalem, Israel, October 14-18, 2013. Proceedings*, volume 8205 of *Lecture Notes in Computer Science*, pages 61–75. Springer, 2013.

[2] G. Asharov, R. Canetti, and C. Hazay. Towards a game theoretic view of secure computation. In K. G. Paterson, editor, *Advances in Cryptology - EUROCRYPT 2011 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings*, volume 6632 of *Lecture Notes in Computer Science*, pages 426–445. Springer, 2011.

[3] K. Chung, F. Liu, C. Lu, and B. Yang. Efficient string-commitment from weak bit-commitment. In M. Abe, editor, *Advances in Cryptology - ASIACRYPT 2010 - 16th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 5-9, 2010. Proceedings*, volume 6477 of *Lecture Notes in Computer Science*, pages 268–282. Springer, 2010.

[4] Y. Dodis and T. Rabin. Cryptography and game theory. In N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, editors, *Algorithmic Game Theory*, pages 181–207. Cambridge University Press, New York, NY, USA, 2007.

[5] G. Fuchsbauer, J. Katz, and D. Naccache. Efficient rational secret sharing in standard communication networks. In Micciancio [19], pages 419–436.

[6] J. A. Garay, J. Katz, U. Maurer, B. Tackmann, and V. Zikas. Rational protocol design: Cryptography against incentive-driven adversaries. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 648–657. IEEE Computer Society, 2013.

[7] J. A. Garay, J. Katz, B. Tackmann, and V. Zikas. How fair is your protocol?: A utility-based approach to protocol optimality. In C. Georgiou and P. G. Spirakis, editors, *Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing, PODC 2015, Donostia-San Sebastián, Spain, July 21 - 23, 2015*, pages 281–290. ACM, 2015.

[8] O. Goldreich. *The Foundations of Cryptography - Volume 2, Basic Applications*. Cambridge University Press, 2004.

[9] S. Goto and J. Shikata. A compiler of two-party protocols for composable and game-theoretic security, and its application to oblivious transfer. In J. Groth, editor, *Cryptography and Coding - 15th IMA International Conference, IMACC 2015, Oxford, UK, December 15-17, 2015. Proceedings*, volume 9496 of *Lecture Notes in Computer Science*, pages 133–151. Springer, 2015.

[10] R. Gradwohl. Rationality in the full-information model. In Micciancio [19], pages 401–418.

[11] R. Gradwohl, N. Livne, and A. Rosen. Sequential rationality in cryptographic protocols. *ACM Trans. Economics and Comput.*, 1(1):2, 2013.

[12] A. Groce and J. Katz. Fair computation with rational players. In D. Pointcheval and T. Johansson, editors, *Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings*, volume 7237 of *Lecture Notes in Computer Science*, pages 81–98. Springer, 2012.

[13] A. Groce, J. Katz, A. Thiruvengadam, and V. Zikas. Byzantine agreement with a rational adversary. In A. Czumaj, K. Mehlhorn, A. M. Pitts, and R. Wattenhofer, editors, *Automata, Languages, and Programming - 39th International Colloquium, ICALP 2012, Warwick, UK, July 9-13, 2012, Proceedings, Part II*, volume 7392 of *Lecture Notes in Computer Science*, pages 561–572. Springer, 2012.

[14] J. Y. Halpern and V. Teague. Rational secret sharing and multiparty computation: extended abstract. In L. Babai, editor, *Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, June 13-16, 2004*, pages 623–632. ACM, 2004.

[15] C. Hazay and Y. Lindell. *Efficient Secure Two-Party Protocols - Techniques and Constructions*. Information Security and Cryptography. Springer, 2010.

[16] H. Higo, K. Tanaka, A. Yamada, and K. Yasunaga. A game-theoretic perspective on oblivious transfer. In W. Susilo, Y. Mu, and J. Seberry, editors, *Information Security and Privacy - 17th Australasian Conference, ACISP 2012, Wollongong, NSW, Australia, July 9-11, 2012. Proceedings*, volume 7372 of *Lecture Notes in Computer Science*, pages 29–42. Springer, 2012.

[17] H. Higo, K. Tanaka, and K. Yasunaga. Game-theoretic security for bit commitment. In K. Sakiyama and M. Terada, editors, *Advances in Information and Computer Security - 8th International Workshop on Security, IWSEC 2013, Okinawa, Japan, November 18-20, 2013, Proceedings*, volume 8231 of *Lecture Notes in Computer Science*, pages 303–318. Springer, 2013.

[18] J. Katz. Bridging game theory and cryptography: Recent results and future directions. In R. Canetti, editor, *Theory of Cryptography, Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19-21, 2008.*, volume 4948 of *Lecture Notes in Computer Science*, pages 251–272. Springer, 2008.

[19] D. Micciancio, editor. *Theory of Cryptography, 7th Theory of Cryptography Conference, TCC 2010, Zurich, Switzerland, February 9-11, 2010. Proceedings*, volume 5978 of *Lecture Notes in Computer Science*. Springer, 2010.

[20] R. Pass and A. Shelat. Renegotiation-safe protocols. In B. Chazelle, editor, *Innovations in Computer Science - ICS 2010, Tsinghua University, Beijing, China, January 7-9, 2011. Proceedings*, pages 61–78. Tsinghua University Press, 2011.

[21] K. Yasunaga. Public-key encryption with lazy parties. *IEICE Transactions*, 99-A(2):590–600, 2016.