Information-Theoretic Conditions for Two-Party Secure Function Evaluation

Claude Crépeau[1 *], George Savvides[1 *], Christian Schaffner[2 **], and Jürg Wullschleger[3 ***]

[1] McGill University, Montréal, QC, Canada. {crepeau,gsavvi1}@cs.mcgill.ca
[2] BRICS, University of Århus, Denmark. chris@brics.dk
[3] ETH Zürich, Switzerland. wjuerg@inf.ethz.ch

Abstract. The standard security definition of unconditional secure function evaluation, which is based on the ideal/real model paradigm, has the disadvantage of being overly complicated to work with in practice. On the other hand, simpler ad-hoc definitions tailored to special scenarios have often been flawed. Motivated by this unsatisfactory situation, we give an information-theoretic security definition of secure function evaluation which is very simple yet provably equivalent to the standard, simulation-based definitions.

1 Introduction

1.1 Secure Function Evaluation

Secure function evaluation is a cryptographic task originally introduced by Yao in [30]. In essence, this task enables a set of mutually distrustful parties without access to a trusted intermediary to jointly compute the output of a function f without any party revealing any information about its input or output to the other parties beyond what these parties can infer from their own inputs and outputs. Goldreich, Micali and Wigderson [21] showed how to achieve this for any function f in a computationally secure way. Schemes ensuring unconditional security were subsequently provided by Ben-Or, Goldwasser and Wigderson [3] and independently by Chaum, Crépeau and Damgård [12].

Micali and Rogaway [25] and Beaver [2] proposed formal security definitions for secure function evaluation. Both definitions were inspired by the simulation paradigm used by Goldwasser, Micali and Rackoff [22] to define zero-knowledge proofs of knowledge. In a nutshell, to each real protocol computing f we associate a two-step procedure in an *ideal* model, where each party simply forwards its input to a trusted party which in turn computes f and distributes the relevant outputs to the parties. The real protocol is deemed secure if any adversary attacking the protocol has a counterpart in the ideal model that achieves a

* Supported in part by NSERC, MITACS, and CIAR.
** Supported by the EC-Integrated Project SECOQC, No: FP6-2002-IST-1-506813.
*** Supported by Canada's NSERC, Québec's FQRNT, and Switzerland's SNF.

similar result simply by processing the input prior to forwarding it to the trusted party, and then by processing the output it receives from it. In other words, a protocol is secure if any attack can be simulated in the much more restrictive ideal model. Such protocols secure in *the ideal/real model paradigm* were later shown to be *sequentially composable* in the sense that the composition of two or more secure protocols is itself a secure protocol. The sequential composability of secure protocols was further explored by Canetti [9, 10] and Goldreich [20].

Canetti [11] also defined *universal composability*, an even stronger security requirement that guarantees that protocols satisfying it can be securely composed *concurrently* in any environment. A similar security definition was provided independently by Backes, Pfitzmann and Waidner [1]. Unfortunately, however appealing the properties of these security definitions may be, they are too strong to allow even basic tasks such as bit commitment to be realized without further assumptions. For this reason, we will limit ourselves to the simpler definition given by Goldreich [20].

1.2 Oblivious Transfer

1-out-of-n string oblivious transfer, denoted $\binom{n}{1}$-OT^k, is a primitive that allows a sender Alice to send one of n binary strings of length k to a receiver Bob. The primitive allows Bob to receive the string of his choice while concealing this choice from (possibly dishonest) Alice. On the other hand, the primitive guarantees that (any dishonest) Bob cannot obtain information about more than one of the strings, including partial joint information on two or more strings.

The first variant of oblivious transfer was introduced by Wiesner [28]. Independently, Rabin re-introduced oblivious transfer in [27] and demonstrated its potential as a cryptographic tool. Its applicability to multi-party computation was shown by Even, Goldreich and Lempel in [19]. It has since been proved that oblivious transfer is in fact sufficient by itself to securely compute any function [23]. More completeness results followed in [14], [15] and [24].

1.3 Contributions

The motivation behind our work was to come up with a general information-theoretic security definition to replace the various ad-hoc definitions proposed in the past for specific cryptographic primitives. To this end, we adopt the standard security definition based on the ideal/real model paradigm of Goldreich [20] for computationally-bounded parties, and adapt it to a model where the parties are allowed to be computationally unbounded and to use independent sources of randomness such as channels. We then distill the relevant security properties of the ideal model into a set of information-theoretic conditions, which we use as a basis for constructing our new formal definition of security. We prove that despite its apparent simplicity, our definition is in fact *equivalent* to the original based on the ideal/real model paradigm. We then examine the important special case of oblivious transfer, and show that in this case, the resulting security requirements can be significantly simplified. We also revisit some of the information-theoretic definitions of security used in the past and point out subtle flaws that some of them contain. As an illustration of the usefulness of our definitions, we give a simple proof for the protocol presented in [29] that optimally inverts $\binom{2}{1}$-OT.

1.4 Shortcomings of Previously Proposed Security Definitions for Oblivious Transfer

We revisit some information-theoretic definitions for oblivious transfer that appear in the literature and list some of their shortcomings. Our examples demonstrate that coming up with the 'right' information-theoretic definition is a delicate task, which is the reason why in this paper we aim for a security definition provably equivalent to the standard definition based on the ideal/real model paradigm.

Random Inputs In [18], only oblivious transfer with random inputs is considered, thereby restricting the scope of the proposed definitions to only a few special cases.

Problems with the Security for the Receiver In [5, 26], the definition of security for the receiver requires that the sender's view be independent of the receiver's input. This is often unattainable: in the most general case, where we assume that there is a known dependency between the inputs, no protocol can satisfy the above security condition since the sender's input, which is always part of his own view, will be correlated with the input of the receiver. The definition should instead require that the two variables be independent *given the sender's input.*

Problems with the Security for the Sender The security for the sender is more difficult to formalize correctly. In addition to problems analogous to the ones presented above for the definition of security for the receiver ([5, 26]), there are several commonly encountered difficulties:

- In [6, 16] a dishonest receiver is only allowed to change his input in a *deterministic* way. Specifically, the random variable C' indicating the receiver's *effective input* (i.e., the bit he eventually obtains) must be a deterministic function of the input C, in contrast to the ideal model where C' can be chosen probabilistically by the dishonest receiver.
- In [7] the random variable C' may depend on the honest sender's input, which is impossible in the ideal model. Furthermore, the view V of the dishonest receiver is required to be independent of the honest sender's input X conditioned on the original input C and the receiver's output $X_{C'}$, *but not on* C'. This definition will hence admit some clearly insecure protocols. For example, suppose the dishonest receiver picks $C' \in \{0, 1\}$ uniformly at random (independently of C) and the protocol allows him to output $V = (X_{C'}, X_{1-C'} \oplus C')$. While it is true that V is independent of X_0, X_1 given

C, X'_C, no such protocol can be simulated in the ideal model since both inputs can be deduced from C' and V.

Abort In [6, 16, 7], the honest player is allowed to abort the protocol. However, it is possible that the dishonest player gets some information *before* the honest player aborts, or that the fact of aborting itself provides information about the honest player's inputs.

A correct definition is given in [17] in the context of the bounded-storage model. However, this definition is overly complicated and requires a special *setup stage*, which is in general not present in OT protocols.

1.5 Preliminaries

Let X, Y, and Z be three random variables. We will often use expressions of the form

$$\mathrm{I}(X;Y \mid Z) = 0 \ ,$$

where I is the conditional mutual Shannon information. This means that X *and* Y *are independent, given* Z. The same condition can also be expressed by saying that X*,* Y *and* Z *form a Markov-chain*,

$$X \leftrightarrow Z \leftrightarrow Y \ ,$$

or by

$$P_{Y|ZX} = P_{Y|Z} \ .$$

By the *chain rule* for mutual information we have

$$\mathrm{I}(X;YW \mid Z) = \mathrm{I}(X;W \mid Z) + \mathrm{I}(X;Y \mid WZ) \ .$$

The *information processing inequality* says that local computation cannot increase mutual information. In other words, for any probabilistic f we have

$$\mathrm{I}(X;Y \mid Z) \geq \mathrm{I}(f(X);Y \mid Z) \ .$$

The *statistical distance* or *variational distance* between the distributions of two random variables X and Y over the same domain $\mathcal{V}$ is defined as

$$\delta(X,Y) = \frac{1}{2} \sum_{v \in \mathcal{V}} \big| \Pr[X = v] - \Pr[Y = v] \big| .$$

We also use the notation $X \equiv_\varepsilon Y$ for $\delta(X,Y) \leq \varepsilon$. If X and Y have the *same* distribution, i.e., $\delta(X,Y) = 0$, we write $X \equiv Y$. The statistical distance can alternatively be expressed as:

$$\delta(X,Y) = \max_S \left(\Pr[X \in S] - \Pr[Y \in S] \right) \ .$$

From this expression it is easy to see that the optimal algorithm distinguishing the two distributions can succeed with probability exactly $\frac{1}{2} + \delta(X,Y)$. Another important property of the statistical distance is that for any random variables X and Y, there exists a random variable $\tilde{X}$ with the same distribution as Y satisfying $\Pr[\tilde{X} \neq X] = \delta(X,Y)$.

2 Definition of Secure Function Evaluation

In this section we provide a definition of secure function evaluation. We follow Definition 7.2.10 of [20] (see also [10]) but modify the associated model as follows:

i) We allow the adversary to be *computationally unbounded*.
ii) We require that the output distributions of the ideal and the real model be either *perfectly indistinguishable* or *statistically indistinguishable* (as opposed to computationally indistinguishable).
iii) We consider the input alphabet to be *fixed*.
iv) We allow randomized players that use independent sources of randomness, rather than supplying randomness to otherwise deterministic players.
v) We allow both players to have an output.

Note that ii) and iii) are just consequences of i) while iv) is used to simplify notation and v) simplifies the model by making it symmetric and generalizes it to allow functions such as coin flipping by telephone [4] where both players have an output, but which can be implemented without allowing either party to abort the protocol. In Section 6 we also discuss the model of Definition 7.2.6 of [20], i.e., the model where the first party is allowed to abort the protocol after receiving its result but before the second party receives its own.

We use the following notation: $x \in \mathcal{X}$ denotes the input of the first party, $y \in \mathcal{Y}$ the input of the second party and $z \in \{0,1\}^*$ represents an additional auxiliary input available to both parties but assumed to be ignored by all honest parties. A *g-hybrid protocol* is a pair of (randomized) algorithms $\Pi = (A_1, A_2)$ which can interact by exchanging messages and which additionally have access to the functionality g. More precisely, for a (randomized) function $g : \mathcal{X} \times \mathcal{Y} \to \mathcal{U} \times \mathcal{V}$ the two parties can send x and y to a trusted party and receive u and v, respectively, where $(u, v) = g(x, y)$. Note that a default value is used if a player refuses to send a value. A pair of algorithms $\overline{A} = (\overline{A}_1, \overline{A}_2)$ is called *admissible* for protocol A if either $\overline{A}_1 = A_1$ or $\overline{A}_2 = A_2$, i.e., if at least one of the parties is honest and uses the algorithm defined by the protocol Π.

Definition 1 (Real Model). *Let $\Pi = (A_1, A_2)$ be a g-hybrid protocol and let $\overline{A} = (\overline{A}_1, \overline{A}_2)$ be an admissible pair of algorithms for the protocol Π. The joint execution of Π under $\overline{A}$ on input pair $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and auxiliary input $z \in \{0,1\}^*$ in the real model, denoted by*

$$\text{REAL}^{g}_{\Pi,\overline{A}(z)}(x, y) \ ,$$

is defined as the output pair resulting from the interaction between $\overline{A}_1(x, z)$ and $\overline{A}_2(y, z)$ using the functionality g.

The *ideal model* defines the optimal scenario where the players have access to an ideal functionality f corresponding to the function they wish to compute. A malicious player may therefore only change (1) his input to the functionality and (2) the output he obtains from the functionality.

Definition 2 (Ideal Model). *The trivial f-hybrid protocol $B = (B_1, B_2)$ is defined as the protocol where both parties send their inputs x and y unchanged to the functionality f and output the values u and v received from f unchanged. Let $\overline{B} = (\overline{B}_1, \overline{B}_2)$ be an admissible pair of algorithms for B. The joint execution of f under $\overline{B}$ in the ideal model on input pair $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and auxiliary input $z \in \{0,1\}^*$, denoted by*

$$\text{IDEAL}_{f,\overline{B}(z)}(x, y) ,$$

is defined as the output pair resulting from the interaction between $\overline{B}_1(x, z)$ and $\overline{B}_2(y, z)$ using the functionality f.

Any admissible protocol $\overline{B}$ in the ideal model can be expressed in the following way: the first party receives input (x, z) and the second party receives input (y, z). The two parties produce $(x', z_1) = \overline{B_1^{\text{in}}}(x, z)$ and $(y', z_2) = \overline{B_2^{\text{in}}}(y, z)$, from which x' and y' are inputs to a trusted third party, and z_1 and z_2 are some auxiliary output. The trusted party computes $(u', v') = f(x', y')$ and sends u' to the first party and v' to the second party. The two parties are now given the outputs v' and u' and the auxiliary inputs z_1 and z_2, respectively. The first party outputs $u = \overline{B_1^{\text{out}}}(u', z_1)$ while the second party outputs $v = \overline{B_2^{\text{out}}}(v', z_2)$. Note that if the first party is honest, we have $B_1^{\text{in}}(x, z) = (x, \perp)$ and $B_1^{\text{out}}(u', z_1) = u'$ and similarly for the second party.

Now, to show that a g-hybrid protocol Π securely computes a functionality f, we have to show that anything an adversary can do in the real model can also be done in the ideal model.

Definition 3 (Perfect Security). *A g-hybrid protocol Π* securely computes f perfectly *if for every pair of algorithms $\overline{A} = (\overline{A}_1, \overline{A}_2)$ that is admissible in the real model for the protocol Π, there exists a pair of algorithms $\overline{B} = (\overline{B}_1, \overline{B}_2)$ that is admissible in the ideal model for protocol B (and where the same players are honest), such that for all $x \in \mathcal{X}$, $y \in \mathcal{Y}$, and $z \in \{0,1\}^*$, we have*

$$\text{IDEAL}_{f,\overline{B}(z)}(x, y) \equiv \text{REAL}^g_{\Pi,\overline{A}(z)}(x, y) .$$

It is sometimes not possible to achieve perfect security. The following definition captures the situation where the simulation has a (small) error ε, defined as the maximal statistical distance between the output distributions in the real and ideal model.

Definition 4 (Statistical Security). *A g-hybrid protocol Π* securely computes f with an error of at most ε *if for every pair of algorithms $\overline{A} = (\overline{A}_1, \overline{A}_2)$ that is admissible in the real model for the protocol Π, there exists a pair of algorithms $\overline{B} = (\overline{B}_1, \overline{B}_2)$ that is admissible in the ideal model for protocol B (and where the same players are honest), such that for all $x \in \mathcal{X}$, $y \in \mathcal{Y}$, and $z \in \{0,1\}^*$, we have*

$$\text{IDEAL}_{f,\overline{B}(z)}(x, y) \equiv_\varepsilon \text{REAL}^g_{\Pi,\overline{A}(z)}(x, y) .$$

The statistical distance is used because it has nice properties and intuitively measures the error of a computation: a protocol Π which securely computes f with an error of at most ε, computes f *perfectly* with probability at least $1 - \varepsilon$.

A very important property of the above definitions is that they imply *sequential composition.* The following theorem has been proven in [10].

Theorem 1. *If an h-hybrid protocol Γ securely computes g with an error of at most γ and a g-hybrid protocol Π securely computes f with an error of at most π, then the composed protocol Π^Γ, namely the protocol Π where every call to g is replaced by Γ, is an h-hybrid protocol that securely computes f with an error of at most $\pi + t\gamma$, where t is the number of calls of Π to g.*

2.1 Efficient Simulation

So far, we have not been talking about *efficiency.* Indeed, if we live in a world where every participant has unlimited computer power, efficiency is not an issue, and our security definitions work well. In the world of zero-knowledge interactive proof systems [22] we have learned that "perfect zero-knowledge" is a more powerful notion than "zero-information" because the former also imposes computational conditions that require an efficient simulator. In this paper we choose to focus on the latter because in the context of two-party secure function evaluation, even in the simplest case security is not yet properly defined. When considering computationally bounded adversaries, the situation is different: It might be the case that, even though an attack in the ideal model is possible in principle, simulation is infeasable, because it takes much more time than to attack the real protocol. This problem can be solved by requiring that the running time of the ideal adversary is polynomial in the running time of the real adversary. We do not consider efficient simulation any further in this paper.

3 Secure Function Evaluation from an Information-Theoretic Point of View

In this section, we adopt an information-theoretic view of the security definition. We change our notation slightly to make it more suitable to the information-theoretic domain. We let X, Y and Z be random variables denoting the inputs, distributed according to an unknown distribution. Likewise, we let U and V be random variables denoting the outputs of the two parties. Hence, for specific inputs x, y, z we have

$$(U, V) = \text{REAL}^g_{\Pi,\overline{A}(z)}(x, y)$$

and

$$(\overline{U}, \overline{V}) = \text{IDEAL}_{f,\overline{B}(z)}(x, y) \ .$$

Note that the condition of Definition 3, namely that for all $x \in \mathcal{X}$, $y \in \mathcal{Y}$, and $z \in \{0, 1\}^*$, we have

$$\text{IDEAL}_{f,\overline{B}(z)}(x, y) \equiv \text{REAL}^g_{\Pi,\overline{A}(z)}(x, y) \ ,$$

can equivalently be expressed as

$$P_{UV|XYZ} = P_{\overline{UV}|XYZ} \, .$$

We now state our main theorem. It gives an information-theoretic condition for the security of a real protocol, *without the use of an ideal model*. Intuitively, the security condition for player 1 (and its counterpart for player 2) says the following: Since $\mathrm{I}(X; Y' \mid ZY) = 0$, we have $P_{Y'|YZX} = P_{Y'|YZ}$. Therefore, Y' could have been created without knowing X. The condition

$$P_{UV'|XY'YZ}(u, v' \mid x, y', y, z) = \Pr[(u, v') = f(x, y')]$$

ensures that the distributions of U and V' are the same as those of the outputs of f on input X and Y'. Finally, $\mathrm{I}(UX; V \mid ZYY'V') = 0$ ensures that V could have been constructed out of Z, Y, Y' and V', without the help of X and U. Therefore, these conditions ensure that the resulting distribution in the real model could also have been obtained in the ideal model.

Theorem 2. *A g-hybrid protocol Π* securely computes f perfectly *if and only if for every pair of algorithms $\overline{A} = (\overline{A}_1, \overline{A}_2)$ that is admissible in the real model for the protocol Π and for all inputs (X, Y) and auxiliary input Z, $\overline{A}$ produces outputs (U, V), such that the following conditions are satisfied:*

- (Correctness) *If both players are honest, we have*

$$P_{UV|XYZ}(u, v, x, y, z) = \Pr[(u, v) = f(x, y)] \, .$$

- (Security for Player 1) *If player 1 is honest, then there exist random variables Y' and V' such that we have*

$$\mathrm{I}(X; Y' \mid ZY) = 0 \, ,$$

$$P_{UV'|XY'YZ}(u, v' \mid x, y', y, z) = \Pr[(u, v') = f(x, y')] \, ,$$

and

$$\mathrm{I}(UX; V \mid ZYY'V') = 0 \, .$$

- (Security for Player 2) *If player 2 is honest, then there exist random variables X' and U', such that we have*

$$\mathrm{I}(Y; X' \mid ZX) = 0 \, ,$$

$$P_{U'V|X'YXZ}(u', v \mid x', y, x, z) = \Pr[(u', v) = f(x', y)] \, ,$$

and

$$\mathrm{I}(VY; U \mid ZXX'U') = 0 \, .$$

Proof. Let us first assume that the protocol Π securely computes f. Then there exists an admissible pair of algorithms $\overline{B} = (\overline{B}_1, \overline{B}_2)$ for the ideal model such that for all $x \in \mathcal{X}$, $y \in \mathcal{Y}$, and $z \in \{0,1\}^*$, we have

$$\text{IDEAL}_{f,\overline{B}(z)}(x,y) \equiv \text{REAL}^{g}_{\Pi,\overline{A}(z)}(x,y) \ ,$$

or equivalently,

$$P_{UV|XYZ} = P_{\overline{UV}|XYZ} \ .$$

If both players are honest we have $\overline{B} = B$. B_1 and B_2 forward their inputs (X,Y) unchanged to the trusted third party, get back $(\overline{U}', \overline{V}') := f(X,Y)$ and output $(\overline{U}, \overline{V}) = (\overline{U}', \overline{V}')$. This establishes the correctness condition.

Without loss of generality, let player 1 be honest and player 2 be malicious. Let us look at the execution of $\overline{B} = (B_1, \overline{B}_2)$. The malicious $\overline{B}_2$ can be modeled by the two conditional probability distributions $P_{\overline{Y}'\overline{Z}_2|YZ}$ computing the input to the ideal functionality and some internal data $\overline{Z}_2$, and $P_{\overline{V}|\overline{V}'\overline{Z}_2}$ computing the output. Note that we can write $P_{\overline{Y}'\overline{Z}_2|YZ} = P_{\overline{Y}'|YZ} P_{\overline{Z}_2|YZ\overline{Y}'}$, i.e., we can say that $\overline{Y}'$ is computed from X and Z, and that $\overline{Z}_2$ is computed from Y, Z, and $\overline{Y}'$. Clearly, we have

$$\mathrm{I}(X; \overline{Y}' \mid ZY) = 0 \ .$$

The honest B_1 always sends X to the trusted party, which computes $(\overline{U}', \overline{V}') = f(X, \overline{Y}')$ and sends the results to B_1 and $\overline{B}_2$. Since B_1 always outputs $\overline{U} = \overline{U}'$, we have

$$P_{\overline{UV}'|X\overline{Y}'YZ}(u, v' \mid x, y', y, z) = \Pr[(u, v') = f(x, y')] \ .$$

$\overline{B}_2$'s output $\overline{V}$ only depends on $\overline{V}'$ and $\overline{Z}_2$, which only depends on Y, Z and $\overline{Y}'$. It follows that

$$\mathrm{I}(\overline{U}X; \overline{V} \mid ZY\overline{Y}'\overline{V}') = 0 \ .$$

Since the probability distributions $P_{UV|XYZ}$ and $P_{\overline{UV}|XYZ}$ are identical, there must exist random variables satisfying the same properties for the output of protocol Π in the real model. Consequently, there must exist random variables Y' and V', such that

$$\mathrm{I}(X; Y' \mid ZY) = 0 \ ,$$

$$P_{UV'|XY'YZ}(u, v' \mid x, y', y, z) = \Pr[(u, v') = f(x, y')] \ ,$$

and

$$\mathrm{I}(UX; V \mid ZYY'V') = 0 \ .$$

Now assume that the conditions of Theorem 2 hold. If both players are honest, the correctness condition implies $P_{UV|XYZ} = P_{\overline{UV}|XYZ}$. If both players are malicious nothing needs to be shown. Without loss of generality, let player 1 be honest and player 2 be malicious. We will define an admissible protocol $\overline{B} = (B_1, \overline{B}_2)$ in the ideal model that produces the same distribution as the protocol

Π in the real model. Let $\overline{B}_2$ choose his input $\overline{Y}'$ according to $P_{\overline{Y}'|YZ} := P_{Y'|YZ}$, and let him choose his output $\overline{V}$ according to $P_{\overline{V}|YZ\overline{Y}'\overline{V}'} := P_{V|YZY'V'}$. The conditional distribution of the output in the ideal model is given by

$$P_{\overline{UV}|XYZ} = \sum_{y',v'} P_{\overline{Y}'|YZ} P_{\overline{UV}'|X\overline{Y}'} P_{\overline{V}|YZ\overline{Y}'\overline{V}'} ,$$

where

$$P_{\overline{UV}'|X\overline{Y}'}(u, v' \mid x, y') = \Pr[(u, v') = f(x, y')] .$$

From $\mathrm{I}(X; Y' \mid ZY) = 0$ and $\mathrm{I}(UX; V \mid ZYY'V') = 0$ it follows that $P_{Y'|XYZ} = P_{Y'|YZ}$ and $P_{V|XYZY'UV'} = P_{V|YZY'V'}$. Furthermore, we have $P_{UV'|XY'YZ} = P_{\overline{UV}'|X\overline{Y}'}$. As for the conditional distribution of the output in the real model, we have:

$$\begin{aligned} P_{UV|XYZ} &= \sum_{y',v'} P_{Y'UV'|XYZ} P_{V|XYZY'UV'} \\ &= \sum_{y',v'} P_{Y'|XYZ} P_{UV'|XYZY'} P_{V|YZY'V'} \\ &= \sum_{y',v'} P_{\overline{Y}'|YZ} P_{\overline{UV}'|X\overline{Y}'} P_{\overline{V}|YZ\overline{Y}'\overline{V}'} \\ &= P_{\overline{UV}|XYZ} . \end{aligned}$$

Therefore, for any admissible $\overline{A}$ in the real model there exists an admissible $\overline{B}$ in the ideal model such that

$$\text{IDEAL}_{f,\overline{B}(z)}(x, y) \equiv \text{REAL}^{g}_{\Pi,\overline{A}(z)}(x, y) ,$$

implying that the protocol is perfectly secure. □

Note that the expression

$$P_{UV'|XY'YZ}(u, v' \mid x, y', y, z) = \Pr[(u, v') = f(x, y')]$$

can be replaced by $(U, V') = f(X, Y')$ if f is deterministic. This yields the following corollary for deterministic functionalities.

Corollary 1. *A protocol Π* securely computes the deterministic functionality f perfectly*, if and only if for every pair of algorithms $\overline{A} = (\overline{A}_1, \overline{A}_2)$ that is admissible in the real model for the protocol Π and for all inputs (X, Y) and auxiliary input Z, $\overline{A}$ produces outputs (U, V), such that the following conditions are satisfied:*

- (Correctness) *If both players are honest, we have $(U, V) = f(X, Y)$.*
- (Security for Player 1) *If player 1 is honest then there exist random variables Y' and V' such that $(U, V') = f(X, Y')$,*

$$\mathrm{I}(X; Y' \mid ZY) = 0 , \qquad \textit{and} \qquad \mathrm{I}(UX; V \mid ZYY'V') = 0 .$$

– (Security for Player 2) *If player* 2 *is honest then there exist random variables* X' *and* U' *such that* $(U', V) = f(X', Y)$*,*

$$\mathrm{I}(Y; X' \mid ZX) = 0 , \qquad \textit{and} \qquad \mathrm{I}(VY; U \mid ZXX'U') = 0 .$$

Note that we require the conditions of Theorem 2 and Corollary 1 to hold for all distributions of the inputs (X, Y). In particular, they have to hold for any input distribution $P_{XY|Z=z}$, i.e., given the event that the auxiliary input Z equals z. Since all the requirements are conditioned on Z, it is sufficient to show that the conditions are met for all distributions P_{XY}, ignoring Z in all the expressions.

The information-theoretic security definition of Theorem 2 and Corollary 1 can also be used for protocols which are not perfectly secure. A protocol is *secure with error* ε if for all inputs X, Y, Z, the joint distribution of the outputs has a statistical distance of at most ε from the output of a perfectly secure protocol. In information theory, the distance between distributions is typically expressed using bounds on entropy and mutual information instead of statistical distance. The following inequalities translate such bounds into bounds on statistical distance. Let U be uniformly distributed over the set $\mathcal{X}$.

$$\delta(P_{XYZ}, P_Z P_{X|Z} P_{Y|Z}) \leq \frac{1}{2}\sqrt{2 \ln 2\, \mathrm{I}(X; Y \mid Z)}$$
$$\delta(P_X, P_U) \leq \frac{1}{2}\sqrt{2 \ln 2 (\log |\mathcal{X}| - \mathrm{H}(X))}$$

The first inequality can easily be proved from [13], Lemma 16.3.1 while the second inequality was proved in [8], Lemma 3.4.

4 Oblivious Transfer

We now apply our security definition to 1-out-of-n string oblivious transfer, or $\binom{n}{1}$-OT^k for short. The ideal functionality f_{OT} is defined as

$$f_{\mathrm{OT}}(X, C) := (\perp, X_C) ,$$

where $\perp$ denotes a constant random variable, $X = (X_0, \ldots, X_{n-1})$, $X_i \in \{0,1\}^k$ for $i \in \{1, \ldots, n\}$, and $C \in \{1, \ldots n\}$.

Theorem 3. *A protocol Π securely computes $\binom{n}{1}$-OT^k perfectly if and only if for every pair of algorithms $\overline{A} = (\overline{A}_1, \overline{A}_2)$ that is admissible for protocol Π and for all inputs (X, C) and auxiliary input Z, $\overline{A}$ produces outputs (U, V) such that the following conditions are satisfied:*

– (Correctness) *If both players are honest, then* $(U, V) = (\perp, X_C)$*.*
– (Security for Player 1) *If player* 1 *is honest, then we have* $U = \perp$ *and there exists a random variable* C'*, such that*

$$\mathrm{I}(X; C' \mid ZC) = 0 , \qquad \textit{and} \qquad \mathrm{I}(X; V \mid ZCC'X_{C'}) = 0 .$$

– (Security for Player 2) *If player* 2 *is honest, then we have*

$$\mathrm{I}(C; U \mid ZX) = 0 \; .$$

Proof. We only need to show that the security condition for player 2 is equivalent to the one in Corollary 1:

$$\mathrm{I}(C; X' \mid ZX) + \mathrm{I}(X'_C C; U \mid ZXX') = 0$$

Since X'_C is a function of C and X',

$$\mathrm{I}(X'_C C; U \mid ZXX') = 0 \text{ is equivalent to } \mathrm{I}(C; U \mid ZXX') = 0 \; .$$

From the chain rule it follows that

$$\mathrm{I}(C; X' \mid ZX) + \mathrm{I}(C; U \mid ZXX') = \mathrm{I}(C; X'U \mid ZX) = \mathrm{I}(C; U \mid ZX) + \mathrm{I}(C; X' \mid ZXU) \; .$$

Now choose $X' = (X'_0, \ldots, X'_{n-1})$ as follows: for all values i, let X'_i be chosen according to the distribution $P_{V|ZXU,C=i}$ except for X'_C. We set $X'_C = V$. Note that all X'_i, $0 \leq i \leq n-1$, have distribution $P_{V|ZXU,C=i}$. Thus X' does not depend on C given ZXU, we have $V = X'_C$ and $\mathrm{I}(C; X' \mid ZXU) = 0$. So there always exists a suitable X' [4], and the condition simplifies to $\mathrm{I}(C; U \mid ZX) = 0$. □

The interpretation of these properties of oblivious transfer is quite intuitive: If player 1 is honest, then she can be confident that anything player 2 can do is basically equivalent to choosing a choice bit C' which is possibly different from C. On the other hand, if player 2 is honest, he can be certain that player 1 does not get to know his input C. Theorem 3 shows that in the case of a dishonest sender in $\binom{n}{1}$-OT^k, *privacy alone implies security.* There *always* exists an input X' that a dishonest sender can use in the ideal model to obtain the same results.

5 An Example

In this section we show how the result from the Section 4 can be used to prove the security of a protocol. Our example will be the protocol from [29], where one instance of $\binom{2}{1}$-OT is implemented using one instance of $\binom{2}{1}$-TO, which is an instance of $\binom{2}{1}$-OT in the opposite direction.

Protocol 1 ([29]) *Let player 1 have input* $X = (X_0, X_1) \in \{0,1\} \times \{0,1\}$*, and player 2 have input* $C \in \{0,1\}$*.*

1. *Player 2 chooses* $R \in \{0,1\}$ *at random.*
2. *The two players execute* $\binom{2}{1}$-TO*, where player 1 inputs* $\overline{C} = X_0 \oplus X_1$*, and player 2 inputs* $\overline{X}_0 = R$ *and* $\overline{X}_1 = R \oplus C$*.*
3. *Player 1 receives* $A = \overline{X}_{\overline{C}}$ *and sends* $M = X_0 \oplus A$ *to the player 2.*
4. *Player 1 outputs* $V := R \oplus M$*.*

[4] Note that these values X' are not necessarily known to a malicious player 1.

Theorem 4. *Protocol 1 perfectly securely reduces* $\binom{2}{1}$-OT *to one realization of* $\binom{2}{1}$-TO.

Proof. If both parties are honest, the protocol is correct because we have

$$R \oplus M = R \oplus X_0 \oplus (X_0 \oplus X_1)C \oplus R = X_C \ .$$

Let player 1 be honest, and let $C' := \overline{X}_0 \oplus \overline{X}_1$. Using the data processing inequality,

$$\mathrm{I}(X_0X_1; C' \mid ZC) \leq \mathrm{I}(X_0X_1; \overline{X}_0\overline{X}_1 \mid ZC) \leq \mathrm{I}(X_0X_1; ZC \mid ZC) = 0 \ .$$

Since $M = X_0 \oplus (X_0 \oplus X_1)(\overline{X}_0 \oplus \overline{X}_1) \oplus \overline{X}_0 = X_{C'} \oplus \overline{X}_0$, the values $\overline{X}_0\overline{X}_1M$, $\overline{X}_0C'M$, and $\overline{X}_0C'X_{C'}$ contain the same information. Thus, using the data processing inequality,

$$\begin{aligned} \mathrm{I}(X_0X_1; V \mid ZCC'X_{C'}) &\leq \mathrm{I}(X_0X_1; CZ\overline{X}_0\overline{X}_1M \mid ZCC'X_{C'}) \\ &= \mathrm{I}(X_0X_1; CZ\overline{X}_0C'X_{C'} \mid ZCC'X_{C'}) = 0 \ . \end{aligned}$$

Now let player 2 be honest. Since $A = R \oplus C\overline{C}$ and R is uniform, we have

$$\mathrm{I}(C; U \mid ZX_0X_1) \leq \mathrm{I}(C; X_0X_1ZA \mid ZX_0X_1) = \mathrm{I}(C; A \mid ZX_0X_1) = 0 \ .$$

Thus, the protocol is secure. □

6 Secure Two-Party Computation with Abort

In this section we will briefly discuss the model of Definition 7.2.6 of [20] where the first party is allowed to abort the protocol right after receiving its output but before the second party has received its own. The *ideal model with abort for player* 1 is similar to the ideal model from Definition 2, the only difference being that player 1 is given the option of aborting the computation by sending a bit C to the trusted party after having received his output. The trusted party sends to player 2 the corresponding output if $C = 1$, and $\perp$ if $C = 0$. An honest player always sends $C = 1$. The real model and the definition of security are identical to the definition without abort. We call a protocol that satisfies this definition *secure with abort for player* 1.

Theorem 5. *A g-hybrid protocol Π* securely computes f perfectly with abort for player 1*, if and only if for every pair of algorithms $\overline{A} = (\overline{A}_1, \overline{A}_2)$ that is admissible in the real model for the protocol Π, and for all inputs (X, Y) and auxiliary input Z, $\overline{A}$ produces outputs (U, V), such that the following conditions are satisfied:*

– (Correctness) *If both players are honest, we have*

$$P_{UV|XYZ}(u, v \mid x, y, z) = \Pr[(u, v) = f(x, y)] \ .$$

– (Security for Player 1) *If player* 1 *is honest, then there exist random variables* Y' *and* V'*, such that we have*

$$\mathrm{I}(X; Y' \mid ZY) = 0 ,$$

$$P_{UV'|XY'YZ}(u, v' \mid x, y', y, z) = \Pr[(u, v') = f(x, y')] ,$$

and

$$\mathrm{I}(UX; V \mid ZYY'V') = 0 .$$

– (Security for Player 2) *If player* 2 *is honest, then there exist random variables* X', C *and* U', V'*, such that we have*

$$\mathrm{I}(Y; X' \mid ZX) = 0 ,$$

$$P_{U'V'|X'YXZ}(u', v' \mid x', y, x, z) = \Pr[(u', v') = f(x', y)] ,$$

$$\mathrm{I}(V'Y; UC \mid ZXX'U') = 0 ,$$

and $V = V'$ *if* $C = 1$ *and* $V = \perp$ *if* $C = 0$.

Proof. The proof is identical to that of Theorem 2 for the case where player 1 is honest. We therefore only examine the case where player 2 is honest and player 1 is malicious.

Let us assume that the protocol Π securely computes f. Consequently, there exists an admissible pair of algorithms $\overline{B} = (\overline{B}_1, B_2)$ such that for all $x \in \mathcal{X}$, $y \in \mathcal{Y}$, and $z \in \{0,1\}^*$ we have $P_{UV|XYZ} = P_{\overline{UV}|XYZ}$.

The malicious $\overline{B}_1$ can be modeled by the two conditional probability distributions $P_{\overline{X}'\overline{Z}_2|XZ}$ computing the input to the ideal functionality and some internal data $\overline{Z}_2$, and $P_{\overline{UC}|\overline{U}'\overline{Z}_2}$ computing the output $\overline{U}$ and the bit $\overline{C}$. Note that we can write $P_{\overline{X}'\overline{Z}_2|XZ} = P_{\overline{X}'|XZ} P_{\overline{Z}_2|XZ\overline{X}'}$. Clearly, we have

$$\mathrm{I}(Y; \overline{X}' \mid ZX) = 0 .$$

The ideal functionality computes $\overline{U}', \overline{V}'$ such that

$$P_{\overline{U}'\overline{V}'|\overline{X}'YXZ}(u', v' \mid x', y, x, z) = \Pr[(u', v') = f(x', y)] .$$

B_1 gets back $\overline{U}'$ from the ideal functionality. Based on $X, Z, \overline{X}', \overline{U}'$ he decides to send $\overline{C}$ to the functionality and outputs $\overline{U}$. Hence, we have

$$\mathrm{I}(\overline{V}'Y; \overline{UC} \mid XZ\overline{X}'\overline{U}') = 0 .$$

If $C = 1$, the functionality sends $\overline{V} = \overline{V}'$ to B_2, if $C = 0$ it sends $\overline{V} = \perp$. B_2 outputs $\overline{V}$ unchanged. As $P_{UV|XYZ} = P_{\overline{UV}|XYZ}$ it must be the case that the same conditions hold in the real model, which implies the security condition for player 2.

Now let the conditions of Theorem 5 hold. We define an admissible protocol $\overline{B} = (\overline{B}_1, B_2)$ in the ideal model that produces the same distribution as the protocol Π in the real model. Let $\overline{B}_1$ choose input $\overline{X}'$ according to $P_{\overline{X}'|XZ} := P_{X'|XZ}$, and $(\overline{U}, \overline{C})$ according to $P_{\overline{UC}|XZ\overline{X}'\overline{U}'} := P_{UC|XZX'U'}$. The conditional distribution of the output in the ideal model is given by

$$P_{\overline{UV}|XYZ} = \sum_{x',c,u',v'} P_{\overline{X}'|XZ} P_{\overline{U}'\overline{V}'|\overline{X}'Y} P_{\overline{UC}|XZ\overline{X}'\overline{U}'} P_{\overline{V}|\overline{V}'\overline{C}} ,$$

where

$$P_{\overline{U}'\overline{V}'|\overline{X}'Y}(u', v' \mid x', y) = \Pr[(u', v') = f(x', y)] .$$

From $\mathrm{I}(Y; X' \mid ZX) = 0$ and $\mathrm{I}(V'Y; UC \mid XZX'U') = 0$ it follows that $P_{X'|XYZ} = P_{X'|XZ}$ and $P_{UC|XZX'U'V'Y} = P_{UC|XZX'U'}$. Furthermore, we have $P_{U'V'|X'YXZ} = P_{\overline{U}'\overline{V}'|\overline{X}'Y}$ and $P_{V|V'C} = P_{\overline{V}|\overline{V}'\overline{C}}$. We get for the conditional distribution of the output in the real model

$$\begin{aligned} P_{UV|XYZ} &= \sum_{x',c,u',v'} P_{X'|XYZ} P_{U'V'|XYZX'} P_{UCV|XYZX'U'V'} \\ &= \sum_{x',c,u',v'} P_{X'|XZ} P_{\overline{U}'\overline{V}'|\overline{X}'Y} P_{UC|XYZX'U'V'} P_{V|XYZX'U'V'CU} \\ &= \sum_{x',c,u',v'} P_{X'|XZ} P_{\overline{U}'\overline{V}'|\overline{X}'Y} P_{UC|XZX'U'} P_{V|V'C} \\ &= \sum_{x',c,u',v'} P_{\overline{X}'|XZ} P_{\overline{U}'\overline{V}'|\overline{X}'Y} P_{\overline{UC}|XZ\overline{X}'\overline{U}'} P_{\overline{V}|\overline{V}'\overline{C}} \\ &= P_{\overline{UV}|XYZ} . \end{aligned}$$

Therefore for any admissible $\overline{A}$ in the real model there exists an admissible $\overline{B}$ in the ideal model such that

$$\text{IDEAL}_{f,\overline{B}(z)}(x, y) \equiv \text{REAL}^{g}_{\Pi,\overline{A}(z)}(x, y) ,$$

which means that the protocol is perfectly secure with abort for player 1. □

7 Conclusion and Open Problems

We have shown that various information-theoretic security definitions for oblivious transfer used in the past contain subtle flaws. We propose a new information-theoretic security definition which is provably equivalent to the security definition based on the ideal/real model paradigm. This not only provides a solid security foundation for most protocols in the literature, which turn out to meet our requirements, but also shows that they are in fact sequentially composable.

An interesting open problem is to generalize our model to various quantum settings, for example to the scenario where two players connected by a quantum channel wish to securely implement a classical functionality.

8 Acknowledgements

We thank Abdul Ahsan, Serge Fehr and Stefan Wolf for many helpful discussions and the anonymous referees for their comments.

References

1. M. Backes, B. Pfitzmann, and M. Waidner. A universally composable cryptographic library. Cryptology ePrint Archive, Report 2003/015, 2003.
2. D. Beaver. Foundations of secure interactive computing. In *Advances in Cryptology: CRYPTO '91*, pages 377–391, London, UK, 1992. Springer-Verlag.
3. M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing (STOC '88)*, pages 1–10. Springer-Verlag, 1988.
4. Manuel Blum. Coin flipping by telephone a protocol for solving impossible problems. *SIGACT News*, 15(1):23–27, 1983.
5. C. Blundo, P. D'Arco, A. De Santis, and D. R. Stinson. New results on unconditionally secure distributed oblivious transfer. In *SAC '02: Revised Papers from the 9th Annual International Workshop on Selected Areas in Cryptography*, pages 291–309, London, UK, 2003. Springer-Verlag.
6. G. Brassard, C. Crépeau, and M. Santha. Oblivious transfers and intersecting codes. *IEEETIT: IEEE Transactions on Information Theory*, 42, 1996.
7. G. Brassard, C. Crépeau, and S. Wolf. Oblivious transfers and privacy amplification. *Journal of Cryptology: the journal of the International Association for Cryptologic Research*, 16(4):219–237, 2003.
8. C. Cachin. *Entropy Measures and Unconditional Security in Cryptography*. PhD thesis, No. 12187, ETH Zurich, Switzerland, 1997.
9. R. Canetti. *Studies in Secure Multiparty Computation and Applications*. PhD thesis, Weizmann Instiitute of Science, Israel, 1996.
10. R. Canetti. Security and composition of multiparty cryptographic protocols. *Journal of Cryptology*, 13(1):143–202, 2000.
11. R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. Cryptology ePrint Archive, Report 2000/067, 2000.
12. D. Chaum, C. Crépeau, and I. Damgård. Multiparty unconditionally secure protocols (extended abstract). In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing (STOC '88)*, pages 11–19. ACM Press, 1988.
13. T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, New York, USA, 1991.
14. C. Crépeau. Verifiable disclosure of secrets and applications (abstract). In *Advances in Cryptology: EUROCRYPT '89*, Lecture Notes in Computer Science, pages 181 – 191. Springer-Verlag, 1990.
15. C. Crépeau, J. van de Graaf, and A. Tapp. Committed oblivious transfer and private multi-party computation. In *Advances in Cryptology: CRYPTO '95*, Lecture Notes in Computer Science, pages 110–123, 1995.
16. P. D'Arco and D. R. Stinson. Generalized zig-zag functions and oblivious transfer reductions. In *SAC '01: Revised Papers from the 8th Annual International Workshop on Selected Areas in Cryptography*, pages 87–102, London, UK, 2001. Springer-Verlag.

17. Y. Ding, D. Harnik, A. Rosen, and R. Shaltiel. Constant-round oblivious transfer in the bounded storage model. In *In Theory of Cryptography — TCC '04*, volume 2951. Springer-Verlag, 2004.
18. Y. Dodis and S.Micali. Lower bounds for oblivious transfer reductions. In *Advances in Cryptology: EUROCRYPT '97*, 1999.
19. S. Even, O. Goldreich, and A. Lempel. A randomized protocol for signing contracts. *Commun. ACM*, 28(6):637–647, 1985.
20. O. Goldreich. *Foundations of Cryptography*, volume II: Basic Applications. Cambridge University Press, 2004.
21. O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing (STOC '87)*, pages 218–229. ACM Press, 1987.
22. S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof systems. *SIAM J. Comput.*, 18(1):186–208, 1989.
23. J. Kilian. Founding cryptography on oblivious transfer. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, pages 20–31, 1988.
24. J. Kilian. More general completeness theorems for secure two-party computation. In *STOC*, pages 316–324, 2000.
25. S. Micali and P. Rogaway. Secure computation (abstract). In *Advances in Cryptology: CRYPTO '91*, pages 392–404, London, UK, 1992. Springer-Verlag.
26. V. Nikov, S. Nikova, B. Preneel, and J. Vandewalle. On unconditionally secure distributed oblivious transfer. In *Progress in Cryptology - INDOCRYPT 2002*, pages 395–408, 2002.
27. M. O. Rabin. How to exchange secrets by oblivious transfer. Technical Report TR-81, Harvard Aiken Computation Laboratory, 1981.
28. S. Wiesner. Conjugate coding. *SIGACT News*, 15(1):78–88, 1983.
29. S. Wolf and J. Wullschleger. Oblivious transfer is symmetric. In *Advances in Cryptology: EUROCRYPT '06*, Lecture Notes in Computer Science. Springer-Verlag, 2006.
30. A. C. Yao. Protocols for secure computations. In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science (FOCS '82)*, pages 160–164, 1982.