Sequentially Composable Information Theoretically Secure Oblivious Polynomial Evaluation

Rafael Tonicelli[1], Rafael Dowsley[1], Goichiro Hanaoka[2], Hideki Imai[2], Jörn Müller-Quade[4], Akira Otsuka[2], Anderson C. A. Nascimento[1]

[1] Department of Electrical Engineering, University of Brasilia
Campus Darcy Ribeiro, 70910-900, Brasilia, DF, Brazil
E-mail: {tonicelli, rafaeldowsley}@redes.unb.br, andclay@ene.unb.br
[2] National Institute of Advanced Industrial Science and Technology (AIST)
1-18-13, Sotokanda, Chyioda-ku, 101-0021, Tokyo, Japan
E-mail: {hanaoka-goichiro, h-imai, a-otsuka}@aist.go.jp
[3] Universität Karlsruhe, Institut für Algorithmen und Kognitive Systeme
Am Fasanengarten 5, 76128 Karlsruhe, Germany
E-mail: muellerq@ira.uka.de

Abstract. Oblivious polynomial evaluation (OPE) consists of a two-party protocol where a sender inputs a polynomial P, and a receiver inputs a single value i. At the end of the protocol, the sender learns nothing and the receiver learns $P(i)$. This paper deals with the problem of oblivious polynomial evaluation under an information-theoretical perspective, which is based on recent definitions of Unconditional Security developed by Crépeau et al. [6]. In this paper, we propose an information-theoretical model for oblivious polynomial evaluation relying on pre-distributed data, and prove very general lower bounds on the size of the pre-distributed data, as well as the size of the communications in any protocol. It is demonstrated that these bounds are tight by obtaining a round-optimal OPE protocol, which meets the lower bounds simultaneously. Some applications of the proposed model are provided, such as solutions for the "Millionaire Problem" and the "Oblivious Equality Testing Problem". We also present a natural generalization to OPE called oblivious linear functional evaluation.

1 Introduction

1.1 Secure Function Evaluation

Assume that there are n players, $1, \ldots, n$; each player i has a private input x_i, which is known only to him/her. Their goal is to collaboratively compute $f(x_1, \ldots, x_n)$ in such a way that no player has to reveal unnecessary information about his/her input. A protocol allowing two or more parties to achieve this goal and satisfying both the correctness and the privacy constraints is called a *secure function evaluation protocol*. The correctness constraint implies that the values

the protocol returns are correct, even if some part in the system fails (i.e., the fault part can choose his/her input to the function, but cannot force the protocol to output a wrong value as the result of the function evaluation); and the privacy constraint implies that the joint computation of $f(x_1, \ldots, x_n)$ does not reveal to each participant i more information that can be deduced from $f(x_1, \ldots, x_n)$ and his/her own private input x_i.

There are two main ways of defining the security of a cryptographic system: *Information-Theoretic Security* (also called *Unconditional Security*) and *Computational Security*. For the former no assumption is assumed about the computational power of the adversary. For the later, the security is defined in terms of an adversary with limited computational power. In order to prove that a system is computationally secure, it is necessary to invoke certain unproven intractability assumptions, e.g., the hardness of computing the discrete logarithm. In contrast, information-theoretically secure systems do not rely on any hypothesis about the complexity of certain problems, but rely on physical assumptions, e.g., the existence of noisy channels. In spite of being considered less practical, information-theoretic security is a permanent and stronger definition of security. This work focuses on the oblivious polynomial evaluation problem from an information-theoretical point of view, and is based on formal definitions of unconditionally secure evaluation schemes proposed by Crépeau at al. [6] which corrected many drawbacks present in several ad-hoc definitions of security proposed in the past.

1.2 Oblivious Transfer

Oblivious Transfer (OT), a cryptographic primitive introduced by Rabin [13], is of particular interest in secure multi-party computation. It has been proven that any function can be calculated unconditionally securely (without considering fairness) if oblivious transfer is available [10]. This property is called *completeness*.

A useful variant of oblivious transfer is *one-out-of-n string OT*, denoted by $\binom{n}{1}$-OT^k. It allows a sender to send n strings ($x_0, \ldots, x_{n-1}$) of length k to a receiver, who is allowed to learn one of them according to his choice c. This process is illustrated below.

An OT protocol is said to be correct, if for honest players, the receiver obtains the desired output x_c and both players do not abort the protocol. It is said to be private if the sender learns no information on the receiver's choice c, while the receiver gets information concerning at most one of the sender's inputs.

1.3 Oblivious Polynomial Evaluation

Oblivious Polynomial Evaluation (OPE) is a variant of Oblivious Function Evaluation and was introduced in [12]. Like OT, OPE is a very useful tool for achieving secure distributed computations.

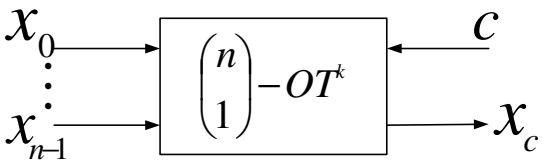

Fig. 1. One-out-of-n string OT.

OPE is a two-party protocol where a sender (Alice) inputs a polynomial over a finite field and a receiver (Bob) inputs a single point of the same finite field. At the end of the protocol, Alice receives nothing and Bob should receive the polynomial input by Alice, evaluated on the point chosen by him. The protocol is secure if Alice learns nothing on which point was chosen by Bob and Bob evaluates the polynomial input by Alice on at most one point.

Fig. 2. Oblivious Polynomial Evaluation. Note that $\mathbf{F}_q$ denotes a finite field.

Since its introduction in [12] OPE has been extensively studied. In [5] the problem of implementing OPE was efficiently reduced to that of achieving OT. Also, in [5] an information theoretically secure protocol for implementing OPE was proposed. The security of that protocol was based on trustiness of a third party which took an active role in the protocol execution.

In this paper, we analyze the problem of achieving information theoretical oblivious polynomial evaluation without using an active (on-line) trusted party.

1.4 Commodity-Based Cryptography

Many security schemes demand an active server for intermediating the interactions among the participants of the protocol. Thus, information exchanged among the participants will depend on the reliability and trustiness of the server during all the protocol execution. One alternative to this is the so-called *Commodity-Based Cryptography*, introduced by Beaver [1].

The protocols proposed in this paper rely on the commodity cryptographic model, where players buy cryptographic primitives from "off-line" servers. These primitives can be used later on to implement general cryptographic protocols.

The commodity based model was inspired on the Internet architecture, which is usually based on the "client-server" paradigm. Once the primitives, or commodities as they are called by Beaver, are acquired, no further interactions between server and users are required. Therefore, the servers need not to know the values which are computed by the players. Moreover, if several servers are available, they need not to be completely trusted, that is, the system is secure even if some servers collude with the users of the commodities and/or among themselves. Another interesting feature of Beaver's model is that no interaction among the servers is required.

In this contribution, we show that the use of "off-line" servers provides very efficient and simple protocols for secure oblivious polynomial evaluation over a finite field.

Although this model was formalized just in [1], several independent works share the same flavor. We cite key-pre-distribution schemes [11], unconditionally secure bit commitments [14, 4] and unconditionally secure digital signature schemes [9].

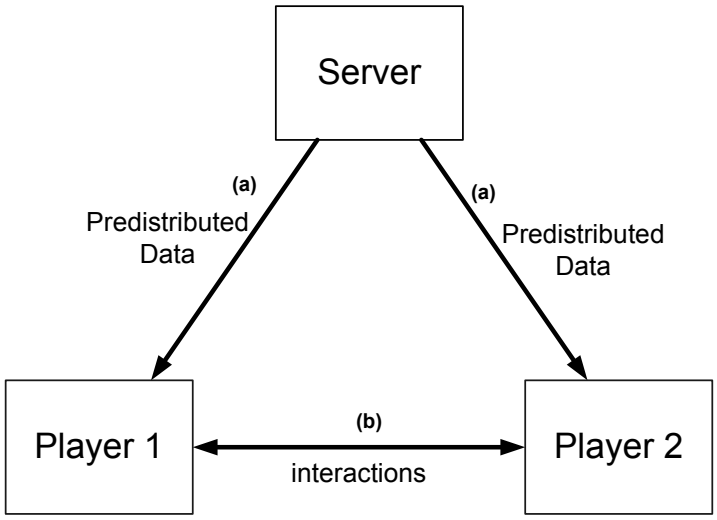

Fig. 3. Process **(a)** represents the setup phase and process **(b)** represents the interactions where no further intervention of the commodity server is needed.

1.5 Contributions and Related Works

Seminal works on secure function evaluation and OPE were computationally secure. For instance, Naor and Pinkas proposed in [12] an OPE scheme which was based on the intractability assumption of *noisy polynomial interpolation*. Later,

Bleichenbacher and Nguyen demonstrated in [3] that this assumption could be less strong than expected and proposed a new intractability assumption based on the *polynomial reconstruction problem*. While the hardness of these problems remains an open question in the foundations of Computer Science, our OPE model is information-theoretically secure, i.e., it is secure even against a computationally unbounded adversary and does not rely on unproven computational hypotheses.

Recently, Crépeau et al. [6] constructed a new formal definition of unconditional security, which is based on the *ideal/real model paradigm*, and established conditions for two-party secure function evaluation in a scenario where the players have infinite computational resources. By proving the security of our model, this work aims at revisiting the problem of oblivious polynomial evaluation from this new information-theoretical point of view.

We propose and solve the problem of implementing information theoretically secure OPE with the help of an off-line party which pre-distributes some data during a setup phase, the so-called commodity-based cryptography model [1]. Our solution is optimal in terms of communication complexity.

We provide a model (section 2), bounds for the amount of memory which is required from players taking part in the protocol (section 3) and a construction which achieves these bounds, thus showing their tightness (section 4).

Finally, we propose a more general protocol called *oblivious linear functional evaluation* (OLF) in section 5. In OLF Alice inputs a linear functional while Bob evaluates this linear functional on a vector of his choice. As a side result of our bounds, we prove the optimality of oblivious transfer protocols proposed by Rivest [14] and Beaver [1]. We also showed that these ideas can be used to solve the "Millionaire Problem" and the "Oblivious Equality Testing Problem".

2 Definitions

In this section, the general OPE model and important definitions used throughout the text are provided. These definitions include the security requirements for OPE realization and the scenarios in which our model is applicable.

2.1 Notation

We denote the random variables using overline and letters (e.g., $\overline{X}$). Its distribution is denote by P_X and its domain by $\mathcal{X}$. X denotes a realization of the random variable $\overline{X}$. When it is necessary, we use subscripted numbers to distinguish different realizations of a random variable. The mutual information of two random variables $\overline{X}$ and $\overline{Y}$ is denoted by $I(\overline{X};\overline{Y})$. The Shannon entropy of a random variable $\overline{X}$ is denoted by $H(\overline{X})$. Similarly, $I(\overline{X};\overline{Y}|\overline{Z})$ and $H(\overline{X}|\overline{Z})$ denote the conditional mutual information and the conditional entropy when conditioned on the random variable $\overline{Z}$.

2.2 Security Definitions

A two-party protocol consists of a program which describes a series of messages to be exchanged and local computations to be performed by the two parties. The protocol is said to halt if no more local computations or message exchanges are required. At the end of an execution of a protocol, each party emits an accept/reject message, depending on the messages he/she received and on the result of local computations.

Defining the security of a two-party protocol, where Oblivious Polynomial Evaluation is an important special case, represents a challenging task. We consider scenarios where the parties are computationally unbounded and the existence of *active* and *passive* adversaries. An active (or malicious) adversary may change his/her behavior arbitrarily and cooperate in order to disrupt the correctness and privacy of the computation. On the other hand, a passive (or semi-honest) adversary is the one who follows the protocol, but may try to acquire more information than what he/she is allowed to know.

The definitions for information-theoretically secure two-party function evaluation used in this text are strongly related to the *real/ideal model paradigm*. In the ideal model, the parties are admitted to have access to a trusted third party, who would receive their private inputs, compute the outcome of the desired functionality f and send to each party the corresponding output. In the real model, no trusted party for computing the functionality f exists (possibly the parties have access to some functionality g), and the mutually distrustful parties should run some protocol to compute f. Intuitively speaking, if the real life protocol can emulate the ideal model, the protocol is said to be secure. In other words, a real life protocol is considered secure, if no adversary can cause more damage in a real execution than an ideal adversary (also known as simulator) can cause in an execution of the ideal protocol. Thus, if a protocol is secure according to this paradigm, an attack against the real life protocol has an effect similar to an attack against the ideal model, where the participants have only a black-box access to the desired functionality.

We shall now define when a protocol perfectly securely evaluates a function $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{U} \times \mathcal{V}$. To accomplish this task, we will use the formalism and definitions of [6] (that are based on those of [8]). Let $X \in \mathcal{X}$ be the input of the first player and $Y \in \mathcal{Y}$ the input of the second player. Consider also an additional auxiliary input $Z \in \{0,1\}^*$ that can be potentially used by both players. An honest player will ignore this additional input. A g-hybrid protocol consists of a pair of algorithms $\Pi = (A_1, A_2)$ that can interact by means of two-way message exchange and have access to some functionality g. A pair of algorithms $\widetilde{A} = (\widetilde{A_1}, \widetilde{A_2})$ is admissible for protocol Π if at least one of the parties is honest, that is, if at least one of the equalities $\widetilde{A_1} = A_1$ and $\widetilde{A_2} = A_2$ is true. Note that no security is required when both parties are dishonest ($(\widetilde{A_1} \neq A_1)$ and $(\widetilde{A_2} \neq A_2)$).

The Real Model. In the real model, the players have no access to a trusted intermediary and must compute the desired functionality by means of a g-hybrid

protocol $\Pi = (A_1, A_2)$. Consider $\widetilde{A} = (\widetilde{A_1}, \widetilde{A_2})$ an admissible pair of algorithms for the protocol Π. The joint execution of Π under $\widetilde{A}$ in the real model,

$$REAL^g_{\Pi,\widetilde{A(Z)}}(X,Y),$$

denotes the resulting output pair, given the input pair $(X,Y) \in \mathcal{X} \times Y$, the auxiliary input Z and the functionality g used by $\widetilde{A}$.

The Ideal Model. In the ideal model, both players have access to a trusted third party to evaluate the functionality f. The trivial protocol $B = (B_1, B_2)$ is the protocol where both parties send their inputs to the functionality f and output the values that the functionality f outputs to them. The algorithms $\widetilde{B_1}$ and $\widetilde{B_2}$ of the protocol $\widetilde{B} = (\widetilde{B_1}, \widetilde{B_2})$ receive the inputs X and Y, respectively, and the auxiliary input Z. The algorithms send the values X' and Y' to the trusted party, who returns the value $(U', V') = f(X', Y')$. Finally, $\widetilde{B_1}$ and $\widetilde{B_2}$ output the values U and V. Let $\widetilde{B} = (\widetilde{B_1}, \widetilde{B_2})$ be an admissible pair of algorithms for B. The joint execution of f under $\widetilde{B}$ in the ideal model on input pair (X,Y) and auxiliary input Z, given by

$$IDEAL_{f,\widetilde{B(Z)}}(X,Y),$$

represents the output pair that results from the interaction between $\widetilde{B_1(X,Z)}$ and $\widetilde{B_2(Y,Z)}$ under the functionality f.

Definition 1 (Perfect Security). *A g-hybrid protocol Π evaluates a function f perfectly securely if for every admissible pair of algorithms $\widetilde{A} = (\widetilde{A_1}, \widetilde{A_2})$ in the real model for the protocol Π, there exists an admissible pair of algorithms $\widetilde{B} = (\widetilde{B_1}, \widetilde{B_2})$ in the ideal model for the trivial protocol B, such that*

$$REAL^g_{\Pi,\widetilde{A(Z)}}(X,Y) \equiv IDEAL_{f,\widetilde{B(Z)}}(X,Y).$$

for all input pair $(X,Y) \in \mathcal{X} \times \mathcal{Y}$ and auxiliary input $Z \in \{0,1\}^*$. Note that the symbol $\equiv$ denotes identical distributions.

Next we present a theorem from [6].

Theorem 1. *A protocol Π is said to securely evaluate the deterministic functionality f perfectly, if and only if for every pair of algorithms $\widetilde{A} = (\widetilde{A_1}, \widetilde{A_2})$ that is admissible in the real model for the protocol Π and for all inputs $(X,Y) \in \mathcal{X} \times \mathcal{Y}$ and for all auxiliary input $Z \in \{0,1\}^*$, $\widetilde{A}$ 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 Alice) If Alice is honest then there exist random variables $\overline{Y'}$ and $\overline{V'}$ such that $(U,V') = f(X,Y')$,*

$$I(\overline{X}; \overline{Y'} | \overline{ZY}) = 0, \text{ and } I(\overline{UX}; \overline{V} | \overline{ZYY'V'}) = 0.$$

– *(Security for Bob) If Bob is honest then there exist random variables* $\overline{X'}$ *and* $\overline{U'}$ *such that* $(U', V) = f(X', Y)$,

$$I(\overline{Y}; \overline{X'}|\overline{ZX}) = 0, \text{ and } I(\overline{VY}; \overline{U}|\overline{ZXX'U'}) = 0.$$

The security definitions are now applied to the Oblivious Polynomial Evaluation problem adapting the random variables denotation to the one that we use henceforth in this paper. The ideal functionality f_{OPE} is denoted by

$$f_{\text{OPE}}(P, i) := (\perp, P(i))$$

such that $i, P(i) \in \mathbb{F}_q$, where $\mathbb{F}_q$ is a finite field, P is a polynomial defined over $\mathbb{F}_q$ and $\perp$ denotes a constant random variable. $\overline{P}$ and $\overline{i}$ can have an arbitrary probability distribution.

Theorem 2. *A protocol* Π *realizes an OPE perfectly securely if and only if for every admissible pair of algorithms* $\widetilde{A} = (\widetilde{A_1}, \widetilde{A_2})$ *for protocol* Π *and for all inputs* (P, i) *and auxiliary input* Z, $\widetilde{A}$ *produce outputs* (U, V) *such that the following conditions are satisfied:*

– *(Correctness) If both players are honest, then* $(U, V) = (\perp, P(i))$
– *(Security for Alice) If Alice is honest, then we have* $U = \perp$ *and there exists a random variable* $\overline{i'}$*, such that*

$$I(\overline{P}; \overline{i'}|\overline{Zi}) = 0, \text{ and } I(\overline{P}; \overline{V}|\overline{Zii'P(i')}) = 0$$

– *(Security for Bob) If Bob is honest, then we have*

$$I(\overline{i}; \overline{U}|\overline{ZP}) = 0$$

Proof. We have to prove the equivalence between the privacy conditions for Bob in Theorems 1 and 2. This proof is analogous to the one presented in [6] for one-out-of-n string OT.

According to Theorem 1, we must have that

$$I(\overline{i}; \overline{P'}|\overline{ZP}) = 0 \text{ and } I(\overline{P(i)i}; \overline{U}|\overline{ZPP'U'}) = 0$$

or equivalently,

$$I(\overline{i}; \overline{P'}|\overline{ZP}) + I(\overline{P'(i)i}; \overline{U}|\overline{ZPP'}) = 0.$$

$P'(i)$ is a function of i and the polynomial P' so

$$\begin{aligned} I(\overline{P'(i)i}; \overline{U}|\overline{ZPP'}) &= I(\overline{i}; \overline{U}|\overline{ZPP'}) + I(\overline{P'(i)}; \overline{U}|\overline{iZPP'}) \\ &= I(\overline{i}; \overline{U}|\overline{ZPP'}). \end{aligned}$$

Then, $I(\overline{P'(i)i}; \overline{U}|\overline{ZPP'}) = 0$ is equivalent to $I(\overline{i}; \overline{U}|\overline{ZPP'}) = 0$.

Applying the chain rule for mutual information we obtain

$$\begin{aligned} I(\overline{i}; \overline{P'}|\overline{ZP}) + I(\overline{i}; \overline{U}|\overline{ZPP'}) &= I(\overline{i}; \overline{P'U}|\overline{ZP}) \\ &= I(\overline{i}; \overline{U}|\overline{ZP}) + I(\overline{i}; \overline{P'}|\overline{ZPU}) \\ &= I(\overline{i}; \overline{U}|\overline{ZP}). \end{aligned}$$

The last equality follows from the fact that $\overline{P'}$ and $\overline{i}$ are independent given $\overline{ZPU}$. Bob has q available inputs belonging to the finite field $\mathbb{F}_q$, which are denoted by $J = (j_0, j_1, \ldots, j_{q-1})$. The set $P'(J) = (P'(j_0), P'(j_1), \ldots, P'(j_{q-1}))$ is obtained by evaluating P' on the q possible inputs. The value $P'(j_k)$ is chosen according to the conditional distribution $P_{V|ZPU,i=j_k}$ except for $P'(i)$. We make $P'(i) = V$ (where V is the output received by Bob). Since all $P'(j_k)$, such that $k \in [0, q-1]$, have distribution $P_{V|ZPU,i=j_k}$, $\overline{P'}$ does not depend on $\overline{i}$ given $\overline{ZPU}$. Mathematically, $V = P'(i)$, $P_{V|ZPU,i} = P_{V|ZPU}$ and $I(\overline{i}; \overline{P'}|\overline{ZPU}) = 0$.

We also consider the security of Oblivious Linear Functional (OLF) evaluation where Bob inputs $w \in W$ (vector space) and Alice inputs a linear functional $l \in W^*$ (the dual vector space of linear functionals on W). $\overline{w}$ and $\overline{l}$ can have an arbitrary probability distribution. The security conditions are analogous to the ones of the previous theorem. The ideal functionality f_{OLF} is denoted by

$$f_{\text{OLF}}(l, w) := (\perp, l(w)).$$

Theorem 3. *A protocol Π realizes an OLF perfectly securely if and only if for every admissible pair of algorithms $\widetilde{A} = (\widetilde{A_1}, \widetilde{A_2})$ for protocol Π and for all inputs (l, w) and auxiliary input Z, $\widetilde{A}$ produce outputs (U, V) such that the following conditions are satisfied:*

- *(Correctness) If both players are honest, then $(U, V) = (\perp, l(w))$*
- *(Security for Alice) If Alice is honest, then we have $U = \perp$ and there exists a random variable $\overline{w'}$, such that*

$$I(\overline{l}; \overline{w'}|\overline{Zw}) = 0, \text{ and } I(\overline{l}; \overline{V}|\overline{Zww'l(w')}) = 0$$

- *(Security for Bob) If Bob is honest, then we have*

$$I(\overline{w}; \overline{U}|\overline{Zl}) = 0$$

Proof. We have to prove the equivalence between the privacy conditions for Bob in Theorems 1 and 3. This proof is analogous to the previous one.

According to Theorem 1, we must have that

$$I(\overline{w}; \overline{l'}|\overline{Zl}) = 0 \text{ and } I(\overline{l(w)w}; \overline{U}|\overline{Zll'U'}) = 0$$

or equivalently,

$$I(\overline{w}; \overline{l'}|\overline{Zl}) + I(\overline{l'(w)w}; \overline{U}|\overline{Zll'}) = 0.$$

$l'(w)$ is a function of w and the polynomial l' so

$$\begin{aligned} I(\overline{l'(w)w}; \overline{U}|\overline{Zll'}) &= I(\overline{w}; \overline{U}|\overline{Zll'}) + I(\overline{l'(w)}; \overline{U}|\overline{wZll'}) \\ &= I(\overline{w}; \overline{U}|\overline{Zll'}). \end{aligned}$$

Then, $I(\overline{l'(w)w}; \overline{U}|\overline{Zll'}) = 0$ is equivalent to $I(\overline{w}; \overline{U}|\overline{Zll'}) = 0$.

Applying the chain rule for mutual information we obtain

$$\begin{aligned} I(\overline{w};\overline{l'}|\overline{Zl}) + I(\overline{w};\overline{U}|\overline{Zll'}) &= I(\overline{w};\overline{l'U}|\overline{Zl}) \\ &= I(\overline{w};\overline{U}|\overline{Zl}) + I(\overline{w};\overline{l'}|\overline{ZlU}) \\ &= I(\overline{w};\overline{U}|\overline{Zl}). \end{aligned}$$

The last equality follows from the fact that $\overline{l'}$ and $\overline{w}$ are independent given $\overline{ZlU}$. For every $w' \in W$ such that $w' \neq w$, the value $l'(w')$ is chosen according to the conditional distribution $P_{V|ZlU,w=w'}$ except for $l'(w)$. We make $l'(w) = V$ (where V is the output received by Bob). Since all $l'(w')$ have distribution $P_{V|ZlU,w=w'}$, $\overline{l'}$ does not depend on $\overline{w}$ given $\overline{ZlU}$. Mathematically, $V = l'(w)$, $P_{V|ZlU,w} = P_{V|ZlU}$ and $I(\overline{w};\overline{l'}|\overline{ZlU}) = 0$.

As mentioned in [6] and proved in [7], it is possible to omit the auxiliary input in the security model, so we will omit it henceforth.

2.3 Commodity-Based OPE

In our model we have three players: Alice, Bob and Ted. We assume the three players to be interconnected by private pairwise channels. The adversary is malicious and may deviate from the original protocol in an arbitrary way. Ted is a trusted center who pre-distributes some secret data to Alice and Bob during a setup phase, but does not take part in the protocol later on. We denote the data received by Alice and Bob by the random variables $\overline{U_a}$ and $\overline{U_b}$. The domains where these data are taken from are denoted by $\mathfrak{U}_a$ and $\mathfrak{U}_b$ respectively. The pre-distributed data are chosen independently of the inputs.

In the computation phase, Alice and Bob interact in order to perform an oblivious polynomial evaluation. We model the probabilistic choices of Alice by a random variable $\overline{R_a}$ and those of Bob by a random variable $\overline{R_b}$, so we can use deterministic functions in the protocol. Note that in this way, all the messages generated by Alice and Bob are well-defined random variables, depending on the polynomial $\overline{P}$ defined over $\mathbb{F}_q$ that Alice chose and on the evaluation point $\overline{i} \in \mathbb{F}_q$ that Bob chose. The protocol can have many rounds of communication. Let the random variables $\overline{e}$ denote all the messages sent by Bob and $\overline{h}$ denote all the messages sent by Alice. As usual, we assume that the messages exchanged by the players and their personal randomness are taken from $\{0,1\}^*$.

We call the view of Alice all the data in her possession, i.e. U_a, R_a, P, e, h and denote it by $View_a$. $View_b$ is defined similarly for Bob. So $\overline{View_a}$ and $\overline{View_b}$ are well-defined random variables.

3 Bounds

In this section we prove bounds for Oblivious Polynomial Evaluation in the commodity based model as specified in the last section. Since we are interested on OPE protocols that can be used with any input probability distribution, we

assume in this section that the input probability distribution has some properties (we assume that $\overline{P}$ and $\overline{i}$ are independently and uniformly chosen) and prove some bounds. In the next section, we present an protocol that meets these bounds and can be used with any input probability distribution.

The following propositions refer to scenarios in which the players follow the protocol (are "honest but curious") and the inputs. We can assume without loss of generality that the output of a corrupted Alice, $\overline{U}$, is her view of the protocol execution and that the output of a corrupted Bob, $\overline{V}$, is his view of the protocol execution.

It is natural to think that, in our scenario, if Bob is given access to Alice's secret data $\overline{U_a}$, he should be able to break the secrecy condition completely, that is he should be able to learn all the information about Alice's input $\overline{P}$. We formally prove this fact in the next proposition.

Proposition 1. *Bob learns all the information on P if he is given access to Alice's pre-distributed data U_a after completing a successful execution of oblivious polynomial evaluation. Mathematically, $H(\overline{P}|\overline{ehU_aU_b}) = 0$.*

Proof. After one successful run of the protocol and after obtaining Alice's data, Bob can try to compute eh for all the possible inputs. The correct input will produce a transcript equal to the one obtained during the protocol execution. Furthermore, the condition of security for Bob states that $I(\overline{i}; \overline{U}|\overline{P}) = 0$. Since $\overline{ehU_a}$ is part of Alice's view, $I(\overline{i}; \overline{ehU_a}|\overline{P}) = 0$ and so

$$H(\overline{i}|\overline{ehU_aP}) = H(\overline{i}|\overline{P}) = H(\overline{i}),$$

where the last step follows from the fact that $\overline{P}$ and $\overline{i}$ are independent. It follows that no two different polynomials should produce the same view, otherwise Alice would obtain knowledge on Bob's inputs (if two polynomials produce the same transcript, Bob's choice must be limited to the points where those polynomials coincide).

An equivalent result holds for Alice: if she is given access to Bob's input and the secret data he received from Ted, she is able to break the protocol's security condition for her completely.

Proposition 2. *Alice learns the point which was chosen by Bob if she is given access to the secret data he received from Ted: $H(\overline{i}|\overline{ehU_aU_b}) = 0$.*

Proof. After the real execution of the protocol is finished, we know from proposition 1 that $H(\overline{P}|\overline{ehU_aU_b}) = 0$. Alice simulates Bob's inputs and determines those that are compatible with the transcript eh. By the security for Alice, there cannot be two different values i_1 and i_2 compatible with the transcript, otherwise the correctness condition would allow Bob to discover $P(i_1)$ and $P(i_2)$ (violating Alice's security). So we have that $H(\overline{i}|\overline{ehU_aU_b}) = 0$.

We now prove another auxiliary result: namely, that the messages exchanged are independent of Alice's and Bob's inputs P and i.

Proposition 3. *In a secure commodity based polynomial evaluation protocol,* $I(\overline{Pi};\overline{eh}) = 0$. *In particular,* $H(\overline{P}|\overline{eh}) = H(\overline{P})$.

Proof. We start by rewriting the mutual information of interest:

$$\begin{aligned} I(\overline{Pi};\overline{eh}) &= I(\overline{PiP(i)};\overline{eh}) \\ &= I(\overline{iP(i)};\overline{eh}) + I(\overline{P};\overline{eh}|\overline{iP(i)}) \\ &= I(\overline{P(i)};\overline{eh}|\overline{i}) + I(\overline{i};\overline{eh}) + I(\overline{P};\overline{eh}|\overline{iP(i)}) \end{aligned}$$

Since the security for Bob states that $I(\overline{i};\overline{U}|\overline{P}) = 0$ and eh is part of Alice's view, we have that $I(\overline{i};\overline{eh}|\overline{P}) = 0$ and so $I(\overline{i};\overline{eh}) = 0$ because $\overline{P}$ is independent of $\overline{i}$. From the security of Alice it follows that $I(\overline{P};\overline{V}|\overline{iP(i)}) = 0$ and since eh is part of Bob's view, we have that $I(\overline{P};\overline{eh}|\overline{iP(i)}) = 0$. Hence we get

$$I(\overline{Pi};\overline{eh}) = I(\overline{P(i)};\overline{eh}|\overline{i}).$$

It remains to prove that the right hand side is 0. Assume this were not the case.

Intuitively, we get a contradiction because $\overline{i}$ is independent of $\overline{eh}$, so Bob could go through the protocol and after receiving h decide which value $P(j)$ he wants to obtain information about. Thus, he could not only learn his allotted $P(i)$ but also some more information, in violation of privacy for Alice.

The formal argument involves our technical condition on the distribution of P. Let $\overline{j} = \overline{i} + 1$; in this way also $\overline{j}$ takes on all values with positive probability, and the first part of our intuitive argument is valid: $I(\overline{P(j)};\overline{eh}|\overline{j}) > 0$, because j can be generated by Bob independently of eh, just as i. Now we can estimate

$$\begin{aligned} I(\overline{P(j)};\overline{j}) &< I(\overline{P(j)};\overline{j}) + I(\overline{P(j)};\overline{eh}|\overline{j}) \\ &= I(\overline{P(j)};\overline{ehj}) \\ &\leq I(\overline{P(j)};\overline{ehijP(i)}) \\ &= I(\overline{P(j)};\overline{ijP(i)}) + I(\overline{P(j)};\overline{eh}|\overline{ijP(i)}) \\ &= I(\overline{P(j)};\overline{j}) + 0, \end{aligned}$$

a contradiction. We have only used standard identities and inequalities, except for the last line: there once more security for Alice was brought to bear, and the independence of $\overline{P(j)}$ and $\overline{P(i)}$ for $i \neq j$.

Hence our assumption was wrong, and the proposition is proved.

Now, we use the above propositions to prove a lower bound on the size of the data which is pre-distributed to Alice.

Theorem 4. *In any commodity based secure polynomial evaluation, the size of the data which is pre-distributed to Alice is as large as the size of the polynomial to be evaluated:* $H(\overline{U_a}) \geq H(\overline{P})$.

Proof. Consider $I(\overline{U_a}; \overline{P}|\overline{ehU_b})$: on the one hand we can rewrite it

$$\begin{aligned} I(\overline{U_a}; \overline{P}|\overline{ehU_b}) &= H(\overline{P}|\overline{ehU_b}) - H(\overline{P}|\overline{ehU_aU_b}) \\ &= H(\overline{P}) - 0, \end{aligned}$$

by propositions 3 and 1 and the fact that $\overline{P}$ is independent of $\overline{U_b}$. On the other hand,

$$I(\overline{U_a}; \overline{P}|\overline{ehU_b}) \leq H(\overline{U_a}|\overline{ehU_b}) \leq H(\overline{U_a}),$$

which, put together with our previous identity, proves the theorem.

Another auxiliary result is actually just a corollary of proposition 3:

Proposition 4. *In any commodity based secure polynomial evaluation protocol,* $H(\overline{iP(i)}|\overline{eh}) = H(\overline{iP(i)}) = H(\overline{i}) + H(\overline{P(i)}|\overline{i})$.

Proof. Proposition 3 states $I(\overline{Pi}; \overline{eh}) = 0$. By data processing, we thus have $I(\overline{P(i)}; \overline{eh}) = 0$, which is just a reformulation of the claim.

Here, we show a bound on the size of the data pre-distributed to Bob.

Theorem 5. *In any commodity based secure polynomial evaluation, the size of the data which is pre-distributed to Bob is bounded by the following expression: for any* $\overline{i} \in \mathbb{F}_q$, $H(\overline{U_b}) \geq H(\overline{i}) + H(\overline{P(i)}|\overline{i})$.

Proof. Consider the following:

$$\begin{aligned} I(\overline{U_b}; \overline{P(i)i}|\overline{ehU_a}) &= H(\overline{P(i)i}|\overline{ehU_a}) - H(\overline{P(i)i}|\overline{ehU_aU_b}) \\ &= H(\overline{i}) + H(\overline{P(i)}|\overline{i}) - 0 \end{aligned}$$

using proposition 4 for the first entropy term, and proposition 2 (plus correctness of the protocol) for the second: i is a function of e, h, U_a and U_b, and all these data together determine the polynomial value $P(i)$. On the other hand,

$$I(\overline{U_b}; \overline{P(i)i}|\overline{ehU_a}) \leq H(\overline{U_b}|\overline{ehU_a}) \leq H(\overline{U_b}),$$

and with the previous identity the claim is proved.

We end this section with bounds on the size of the messages which have to be exchanged between Alice and Bob.

Theorem 6. $H(\overline{e}) \geq H(\overline{i})$ *and* $H(\overline{h}) \geq H(\overline{P})$.

Proof. For the first one, use proposition 2 for the first step in the following chain and then independence of $\overline{i}$ and $\overline{hU_aU_b}$:

$$\begin{aligned} H(\overline{i}) &= I(\overline{i}; \overline{ehU_aU_b}) = I(\overline{i}; \overline{hU_aU_b}) + I(\overline{i}; \overline{e}|\overline{hU_aU_b}) \\ &= I(\overline{i}; \overline{e}|\overline{hU_aU_b}) \leq H(\overline{e}|\overline{hU_aU_b}) \leq H(\overline{e}). \end{aligned}$$

For the second one, use proposition 1 for the first step in the following chain and then independence of $\overline{P}$ and $\overline{ijeU_aU_b}$:

$$\begin{aligned} H(\overline{P}) &= I(\overline{P}; \overline{ehU_aU_b}) = I(\overline{P}; \overline{eU_aU_b}) + I(\overline{P}; \overline{h}|\overline{eU_aU_b}) \\ &= I(\overline{P}; \overline{h}|\overline{eU_aU_b}) \leq H(\overline{h}|\overline{eU_aU_b}) \leq H(\overline{h}). \end{aligned}$$

4 An Optimal Construction

In this section we present a construction based on polynomials over finite fields which matches the lower bounds we proved in the last section and is round optimal, thus proving their tightness. The intuition behind the protocol is that Ted distributes a random evaluation performed on a random polynomial to Alice and Bob during a setup phase. Later on, they will exchange messages to turn the random evaluation into the desired one. The protocol is described below and illustrated in figure 4.

Protocol OPE

Setup Phase: Ted selects uniformly and at random a polynomial $\overline{R(X)} \in \mathbb{F}_q[X]$ of degree n and a point $\overline{d} \in \mathbb{F}_q$. Ted sends $R(X)$ to Alice and $d, g = R(d)$ to Bob.

Computing Phase: Alice's input: $P(X)$ of degree n; Bob's input: $i \in \mathbb{F}_q$

- Bob sends $t = i - d$ to Alice.
- Alice computes $F(X) = P(X + t) + R(X)$ and sends it to Bob.
- Bob computes $F(d) - g = P(d + t) + R(d) - R(d) = P(i)$, the desired output.

Theorem 7. *The above stated protocol is a secure implementation of an oblivious polynomial evaluation protocol. Moreover, it is optimal regarding its space complexity*

Proof. (Correctness) It is easily verifiable the correctness of the protocol. Considering both parties to be honest, we obtain

$$F(d) - g = P(d + t) + R(d) - R(d) = P(i)$$

which proves the correctness property.
(Security for Alice) Let Alice be honest and $i' = d + t$. We have

$$I(\overline{P}; \overline{i'}|\overline{i}) = I(\overline{P}; \overline{d + t}|\overline{i}) = 0$$

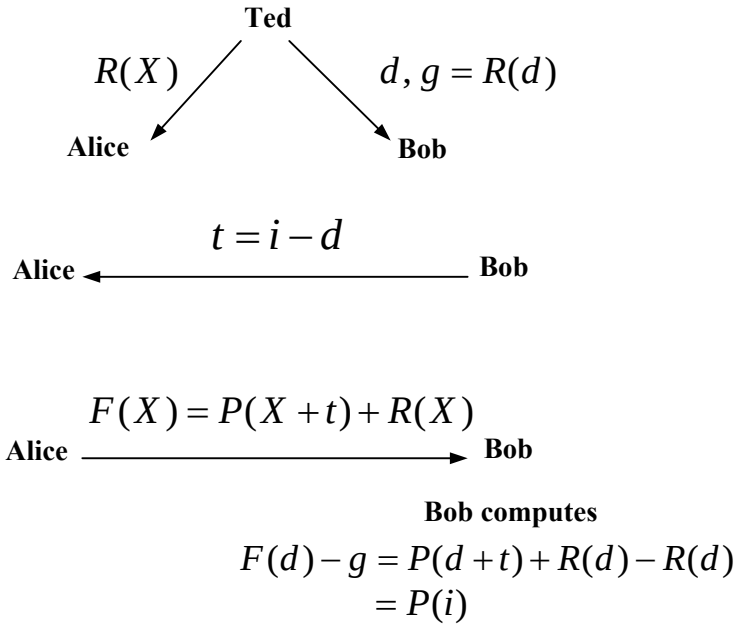

Fig. 4. Optimal OPE protocol construction.

since $\overline{d}$ is independent of $\overline{P}$.
Now we demonstrate that the second condition for Alice is satisfied. We can assume without loss of generality that Bob outputs his view of the protocol execution. We have that

$$\begin{aligned} I(\overline{P};\overline{V}|\overline{ii'P(i')}) &= I(\overline{P};\overline{dR(d)ii'tF}|\overline{ii'P(i')}) \\ &= I(\overline{P};\overline{dR(d)ii'F}|\overline{ii'P(i')}) \\ &= I(\overline{P};\overline{dR(d)F}|\overline{ii'P(i')}) \\ &= I(\overline{P};\overline{F}|\overline{ii'P(i')}) \\ &= 0. \end{aligned}$$

where the first step follows from the fact that t is a function of d and i' and the last step follows from the fact that $F(X) = P(X + t) + R(X)$ where $\overline{R(X)}$ is uniformly random and independent of $\overline{P}$.
(Security for Bob) Let Bob be honest. We can assume without loss of generality that Alice outputs her view of the protocol execution.

$$\begin{aligned} I(\overline{i};\overline{U}|\overline{P}) &= I(\overline{i};\overline{PRtF}|\overline{P}) \\ &= I(\overline{i};\overline{Rt}|\overline{P}) \\ &= I(\overline{i};\overline{R}|\overline{P}) + I(\overline{i};\overline{t}|\overline{PR}) \\ &= 0. \end{aligned}$$

where the first step follows from the fact that F is a function of P, R and t. The last step follows from the fact that the pre-distributed data is independent of the inputs and from the fact that $t = i - d$ (where $\overline{d}$ is uniformly random and independent of $\overline{R}$, $\overline{i}$ and $\overline{P}$).

Finally, our theorems 4, 5 and 6, show that indeed the size of the pre-distributed data as well as of the communicated data meet the lower bounds.

5 Oblivious Linear Functional Evaluation

Here we generalize the previous protocol to the case where Bob inputs $w \in W$ (vector space) and Alice inputs a linear functional $l \in W^*$ (the dual vector space of linear functionals on W). First, notice that evaluating a polynomial $P(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n$ on a point x is the same as evaluating the linear functional $l = (a_0, a_1, \ldots, a_n)$ (as a row vector) on the (column) vector $w = (1, x, x^2, \ldots, x^n)^T$. Thus OPE can be seen as a particular case of oblivious linear functional evaluation. This idea can be generalized to affine linear functionals, but we chose not to break the inherent beautiful symmetry via duality of the problem.

Protocol OLF

Setup Phase: Ted chooses a uniformly random affine linear function $\overline{m}$ and a uniformly random $\overline{d} \in W$. It gives m to Alice and gives d and $c = m(d)$ to Bob.

Computing Phase Alice's input: $l \in W^*$; Bobs input $w \in W$

- Bob sends $t := w - d$ to Alice
- Alice sends the function $n := l + m + l(t)$ to Bob
- Bob computes $n(d) - c = l(d) + m(d) + l(w - d) - m(d) = l(w)$

Theorem 8. *The above stated protocol is a secure implementation of an oblivious linear functional evaluation protocol.*

Proof. (Correctness) It is immediate to verify the correctness of the protocol. Considering both parties to be honest, we obtain

$$n(d) - c = n(d) - m(d) = l(d) + l(w - d) = l(w)$$

(Security for Alice) Let Alice be honest and $w' = d + t$. We have

$$I(\overline{l}; \overline{w'} | \overline{w}) = I(\overline{l}; \overline{d + t} | \overline{w}) = 0$$

since $\overline{d}$ is independent of $\overline{l}$.

Now we demonstrate that the second condition for Alice is satisfied. We can assume without loss of generality that Bob outputs his view of the protocol

execution. We have that

$$\begin{aligned} I(\bar{l}; \overline{V} | \overline{ww'l(w')}) &= I(\bar{l}; \overline{dm(d)ww'tn} | \overline{ww'l(w')}) \\ &= I(\bar{l}; \overline{dm(d)ww'n} | \overline{ww'l(w')}) \\ &= I(\bar{l}; \overline{dm(d)n} | \overline{ww'l(w')}) \\ &= I(\bar{l}; \overline{n} | \overline{ww'l(w')}) \\ &= 0. \end{aligned}$$

where the first step follows from the fact that t is a function of d and w' and the last step follows from the fact that $n = l + m + l(t)$ where $\overline{m}$ is uniformly random and independent of $\bar{l}$.
(Security for Bob) Let Bob be honest. We can assume without loss of generality that Alice outputs her view of the protocol execution.

$$\begin{aligned} I(\overline{w}; \overline{U} | \bar{l}) &= I(\overline{w}; \overline{lmnt} | \bar{l}) \\ &= I(\overline{w}; \overline{mt} | \bar{l}) \\ &= I(\overline{w}; \overline{m} | \bar{l}) + I(\overline{w}; \overline{t} | \overline{lm}) \\ &= 0. \end{aligned}$$

since n is a function of l, m and t. The last step follows from the fact that the pre-distributed data is independent of the inputs and from the fact that $t = w - d$ (where $\overline{d}$ is uniformly random and independent of $\overline{m}$, $\overline{w}$ and $\bar{l}$).

6 Examples

Any function in $\mathbb{F}_q$ can be expressed as a polynomial and so we can evaluate them using Oblivious Polynomial Evaluation, examples of such functions are the Millionaires' and the Oblivious Equality Testing Problems. We present solutions to the Millionaires' and the Oblivious Equality Testing Problems as illustrations of our more general methodology. These solutions are inefficient. Other applications, e.g. the list intersection problem, are presented in [12].

6.1 "Millionaires' Problem"

The "Millionaire's Problem", introduced by Yao in [16], is considered the first problem in secure multi-party computation.

Assume Alice and Bob are two millionaires who want to identify which one is richer, without revealing their actual wealth. Thus, they have to carry out a protocol that allows them to satisfy their curiosity and, simultaneously, their protocol must fulfill the requirements of correctness and privacy.

We show that a single OPE execution gives us a solution to the Millionaires' Problem secure against passive cheating.

Assume that Alice and Bob have values x_a and x_b, respectively, that are members of a subset of the integers $Q = \{0, 1, 2, \ldots, p-2, p-1\}$ (these values

are encoded as numbers in $\mathbb{Z}_p$). To compare the values, Alice chooses $a \in \mathbb{Z}_p$ and generates a function defined over $\mathbb{Z}_p$ of the form

$$P_A(x) = \begin{cases} a, & \text{if } x \geq x_a \\ -a, & \text{if } x < x_a. \end{cases}$$

Alice and Bob run the protocol described in 4, where Alice's input is the function $P_A(x)$ and Bob's input is the value x_b. At the end of the protocol, Bob will receive $P_A(x_b)$.

If $x_b \geqslant x_a$, then $P_A(x_b) = a$.

If $x_b < x_a$, then $P_A(x_b) = -a$.

Then, Bob sends the value $P_A(x_b)$ to Alice.

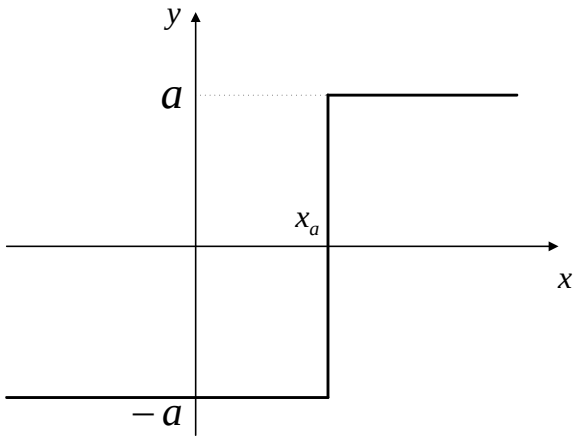

Fig. 5. Function generated to solve the Millionaires' Problem.

6.2 "Oblivious Equality Testing Problem"

In the Oblivious Equality Testing Problem, Alice and Bob want to know if their private inputs x_a and x_b, respectively, are equal without revealing their value. We present a solution based on OPE.

Alice chooses $a \in \mathbb{F}_q$ and generates a function defined over $\mathbb{F}_q$ of the form

$$P_A(x) = \begin{cases} a, & \text{if } x = x_a \\ 0, & \text{if } x \neq x_a. \end{cases}$$

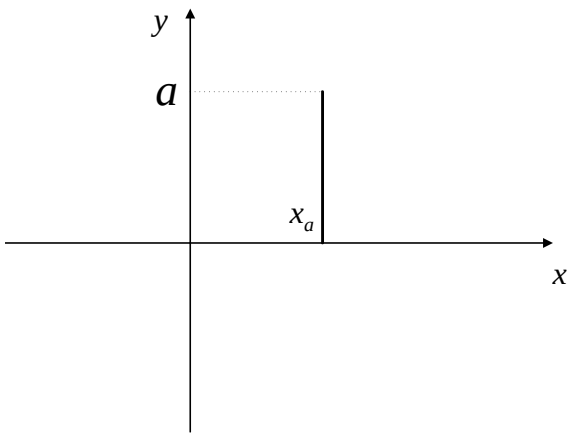

Fig. 6. Function generated to solve the Equivalence Problem.

7 Conclusions

In this paper we introduced and solved the problem of efficiently evaluating polynomials obliviously within the so-called commodity-based cryptography, as proposed by Beaver [1]. We proposed a model and then proved bounds on the amount of "commodities" which have to be pre-distributed by the trusted center, thus providing bounds for the amount of memory required by the players engaged in the protocol, as well as bounds on their communications.

Then, we proved the tightness of our bounds by showing an explicit construction which meets them.

We also presented in this paper a definition of security for oblivious polynomial evaluation which is equivalent to the standard definition based on the *real/ideal model paradigm*. In the light of this new definition, we proved the unconditional security of our schemes.

Finally, we proposed a generalization of oblivious polynomial evaluation: oblivious linear functional evaluation and provided some important applications of OPE.

References

1. D. Beaver. Commodity-Based Cryptography (Extended Abstract). STOC 1997, pp. 446-455, 1997.
2. D. Beaver, S. Micali, and P. Rogaway. The round complexity of secure protocols. In STOC, pages 503513, 1990.
3. D. Bleichenbacher and P. Nguyen, Noisy Polynomial Interpolation and Noisy Chinese Remaindering, Advances in Cryptology Proceedings of EUROCRYPT 2000, LNCS 1807, Springer-Verlag, pp. 5369, 2000.

4. C. Blundo, B. Masucci, D.R. Stinson, and R. Wei. Constructions and Bounds for Unconditionally Secure Non-Interactive Commitment Schemes. Designs, Codes, and Cryptography, Special Issue in Honour of Ron Mullin, 26(1-3), pp. 97-110, 2002.
5. Yan-Cheng Chang, Chi-Jen Lu. Oblivious Polynomial Evaluation and Oblivious Neural Learning. ASIACRYPT 2001, pp. 369-384, 2001.
6. C. Crépeau, G. Savvides, G. Schaffner, J. Wullschleger. Information-theoretic conditions for two-party secure function evaluation. In Advances in Cryptology: EUROCRYPT 2006, Lectures Notes in Computer Science, Springer-Verlag, 2006, pp. 528-554.
7. C. Crépeau, J. Wullschleger: Statistical Security Conditions for Two-Party Secure Function Evaluation. ICITS 2008: 86-99.
8. O. Goldreich. Foundations of Cryptography, volume II: Basic Applications. Cambridge University Press, 2004.
9. G. Hanaoka, J. Shikata, Y. Zheng, and H. Imai. Unconditionally Secure Digital Signature Schemes Admitting Transferability. ASIACRYPT 2000, pp. 130-142, 2000.
10. J. Kilian. Founding Cryptography on Oblivious Transfer. Proc. of 20th ACM Symposium on Theory of Computing (STOC), pp. 20-31, 1988.
11. T. Matsumoto and H. Imai. On the Key Predistribution Systems. A Practical Solution to the Key Distribution Problem. CRYPTO 1987, pp. 185-193, 1988.
12. M. Naor and B. Pinkas. Oblivious transfer and polynomial evaluation. 31st STOC, pp. 245-254, 1999.
13. M. O. Rabin. How to exchange secrets by oblivious transfer. Technical Report TR-81, Harvard, 1981.
14. R.L. Rivest. Unconditionally Secure Commitment and Oblivious Transfer Schemes Using Concealing Channels and a Trusted Initializer. Preprint available from http://theory.lcs.mit.edu/~rivest/Rivest-commitment.pdf
15. S. Wolf and J. Wullschleger. Oblivious transfer is symmetric. In Advances in Cryptology: EUROCRYPT 06, Lecture Notes in Computer Science. Springer-Verlag, 2006.
16. A.C. Yao. Protocols for Secure Computations. FOCS 1982, pp. 160-164, 1982.