On the Efficiency of Bit Commitment Reductions

Samuel Ranellucci[1], Alain Tapp[1], Severin Winkler[2], and Jürg Wullschleger[1,3]

[1] DIRO, Université de Montréal, Quebec, Canada
[2] Institute of Theoretical Computer Science, ETH Zurich, Switzerland
[3] McGill University, Quebec, Canada

Abstract. Two fundamental building blocks of secure two-party computation are *oblivious transfer* and *bit commitment*. While there exist unconditionally secure implementations of oblivious transfer from noisy correlations or channels that achieve constant rates, similar constructions are not known for bit commitment.
In this paper, we show that any protocol that implements n instances of bit commitment with an error of at most 2^{-k} needs at least $\Omega(kn)$ instances of a given resource such as oblivious transfer or a noisy channel. This implies in particular that it is impossible to achieve a constant rate. We then show that it is possible to circumvent the above lower bound by restricting the way in which the bit commitments can be opened. We present a protocol that achieves a constant rate in the special case where only a constant number of instances can be opened, which is optimal. Our protocol implements these restricted bit commitments from string commitments and is universally composable. The protocol provides significant speed-up over individual commitments in situations where restricted commitments are sufficient.

Keywords: secure two-party computation, bit commitment, string commitment, oblivious transfer, noisy channel, information theory

1 Introduction

Commitment schemes [4] are one of the basic building blocks of two-party computation [42]. Commitments can be used in *coin-flipping* [4], *zero-knowledge proofs* [21, 20], *zero-knowledge arguments* [7] or as a tool in general two-party computation protocols to prevent malicious players from actively cheating (see for example [14]).

A commitment scheme has two phases. In the *commit* phase, the sender has to decide on a value b. After the commit phase the value b is fixed and cannot be changed, while the receiver still does not get any information about its value. At a later time, the players may execute the second phase, called the *open* phase, where the bit b is revealed to the receiver. The scheme is called a *bit commitment* if b is only one bit, and it is called a *string commitment* if b is a longer bit string.

Bit commitments can be implemented from a wide variety of information-theoretic primitives [11, 16, 38, 41]. There are protocols which implement a single string commitment from noisy channels at a constant rate, meaning that the size of the string grows linearly with the number of instances of noisy channels used, which is essentially optimal [38]. Protocols which implement individual bit commitments at a constant rate, however, are not known. In [30] it has been shown that in any perfectly correct and perfectly hiding non-interactive bit commitment scheme from distributed randomness with a security of 2^{-k}, the size of the randomness given to the players must be at least $\Omega(k)$.

Another primitive that is of fundamental importance in two-party computation is oblivious transfer (OT) [36, 32, 19]. Oblivious transfer can be implemented from noisy channels [10, 12, 11, 13], cryptogates [28] and weak variants of noisy channels [16, 15, 40, 41]. While all these protocols require $\Omega(k)$ instances of a given primitive to implement a single OT with a security of 2^{-k}, it has been shown in [23, 26, 25, 24] that there are more efficient protocols if many OTs are implemented at once. In the semi-honest model and in some cases also in the malicious model, it is possible to implement OT *at a constant rate*, which means n instances of OT can be implemented from just $O(n)$ instances of the given primitive, if n is big enough compared to the security parameter. It is, therefore, possible to achieve the lower bound for such reductions [17, 2, 39, 37] up to a constant factor. In the following we address the question whether such efficient protocols also exist in the case of bit commitment.

1.1 Contribution

We show that — in contrast to implementations of OT — no constant rate reduction of bit commitment to distributed randomness can exist. More precisely, in Theorem 1 we show that if a protocol implements n bit commitments with a security of at least 2^{-k} from distributed randomness, then the mutual information between the sender's and the receiver's randomness must be almost kn or larger. Our proof is built on the insight that any such protocol must reveal at least k bits of information about the receiver's randomness for each committed bit that is opened. This implies that we need at least $\Omega(kn)$ instances of oblivious transfer or noisy channels to implement n bit commitments. Thus, executing for each bit commitment a protocol that uses $O(k)$ instances is optimal. In combination with the lower bound from [38], this bound can be generalized to string commitments: any protocol that implements n string commitments of length ℓ needs at least $\Omega(n(\ell + k))$ bits of distributed randomness.

However, in many applications of bit commitments the full strength of the commitment scheme is not required. For example in the famous zero-knowledge protocol of [20], it is only required that a constant number of committed bits can be opened. We show that restricting the ways in which the bit commitments can be opened enables us to implement more efficient schemes that circumvent our impossibility result.[4] We introduce a new concept that we call *bit commitments*

[4] Note that for the specific case of zero-knowledge proofs other, more efficient, techniques are known [29].

with restricted openings. It allows a sender to commit to N bits, from which he may open up to $r < N$ one by one. After that, he may only open all the remaining bits at once. Our protocol uses so-called *cover-free families*, and implements bit commitments with restricted openings from string commitments. Together with a simple construction of a cover-free family from [18], our results imply that for any prime power q, we can implement $N = q^2$ bit commitments from which r can be opened from $(r+1)q$ string commitments of length q. (See Corollary 4 for the more general statement.) Together with the protocol from [38], we get a constant-rate bit commitment protocol from noisy channels, for any constant r. As bit commitments with restricted openings are strictly stronger than a string commitment, this is optimal. Together with another construction of a cover-free family from [6], it is possible to implement $N = 2^{\Omega(n/r^2)}$ bit commitments from n string commitments. We prove our protocol secure in the Universal Composability model (UC) [8].

We will prove our lower bounds for independent bit commitments in Section 2. In Section 3, we introduce commitments with restricted openings and give reductions to string commitments. Note that Section 3 can be read without reading Section 2.

1.2 Notation

In the following, the probability distribution of a random variable X is denoted by $P_X(x)$. The joint distribution $P_{XY}(x,y)$ defines a conditional distribution $P_{X|Y}(x,y) = P_{XY}(x,y)/P_Y(y)$ for all y with $P_Y(y) > 0$. The *statistical distance* between the distributions P_X and $P_{X'}$ over the domain $\mathcal{X}$ is defined as

$$\delta(P_X, P_{X'}) := \max_D \mid \Pr[D(X) = 1] - \Pr[D(X') = 1] \mid ,$$

where we maximize over all (inefficient) distinguishers $D : \mathcal{X} \to \{0,1\}$. We use the notation $[n]$ for the set $\{1,\ldots,n\}$. For a sequence $x = (x_1,\ldots,x_n)$ and $t \in [n]$, we denote by x^t the subsequence $(x_1,\ldots,x_t)$.

1.3 Information Theory

We will use the following tools from information theory in our proofs. We assume that the reader is familiar with the basic concepts of information theory, and refer to [9, 22] for more details. The *conditional Shannon entropy* of X given Y is defined as[5]

$$\mathrm{H}(X \mid Y) := -\sum_{x,y} P_{XY}(x,y) \log P_{X|Y}(x,y) \ .$$

We use the notation

$$h(p) = -p\log(p) - (1-p)\log(1-p)$$

[5] All logarithms are binary, and we use the convention that $0 \cdot \log 0 = 0$.

for the binary entropy function, i.e., $h(p)$ is the entropy of the Bernoulli distribution[6] with parameter p. The *mutual information* of X and Y given Z is defined as

$$\mathrm{I}(X;Y \mid Z) = \mathrm{H}(X \mid Z) - \mathrm{H}(X \mid YZ) \ .$$

The mutual information satisfies the following chain rule

$$\mathrm{I}(X_1 \ldots X_n;Y) = \sum_{i=1}^{n} \mathrm{I}(X_i;Y \mid X_1 \ldots X_{i-1}).$$

The Kullback-Leibler divergence or relative entropy of two distributions P_X and Q_X on $\mathcal{X}$ is defined as

$$\mathrm{D}(P_X \parallel Q_X) = \sum_{x \in \mathcal{X}} P_X(x) \log \frac{P_X(x)}{Q_X(x)} \ .$$

The conditional divergence of two distributions P_{XY} and Q_{XY} on $\mathcal{X} \times \mathcal{Y}$ is defined as

$$\mathrm{D}(P_{Y|X} \parallel Q_{Y|X}) = \sum_{x \in \mathcal{X}} P_X(x) \, \mathrm{D}(P_{Y|X=x} \parallel Q_{Y|X=x}) \ .$$

The binary divergence of two probabilities p and q is defined as the divergence of the Bernoulli distributions with parameters p and q, i.e.,

$$\mathrm{d}(p \parallel q) = p \log \frac{p}{q} + (1-p) \log \frac{1-p}{1-q} \ .$$

The divergence (and hence also the conditional divergence) is always non-negative. Furthermore, we have the following chain rule

$$\mathrm{D}(P_{XY} \parallel Q_{XY}) = \mathrm{D}(P_X \parallel Q_X) + \mathrm{D}(P_{Y|X} \parallel Q_{Y|X}) \ . \tag{1}$$

This implies

$$\mathrm{D}(P_X P_{Y|X} \parallel P_X Q_{Y|X}) = \mathrm{D}(P_{Y|X} \parallel Q_{Y|X}). \tag{2}$$

Let Q_X and P_X be two distributions over the inputs to the same channel $P_{Y|X}$. Then the divergence between the outputs $P_Y = \sum_x P_X P_{Y|X}$ and $Q_Y = \sum_x Q_X P_{Y|X}$ of the channel is not greater than the divergence between the inputs, i.e., the divergence satisfies the data-processing inequality

$$\mathrm{D}(P_X \parallel Q_X) \geq \mathrm{D}(P_Y \parallel Q_Y) \ . \tag{3}$$

Furthermore, for random variables X, Y and Z distributed according to P_{XYZ}

$$\mathrm{I}(X;Y \mid Z) = \mathrm{D}(P_{X|YZ} \parallel P_{X|Z}) \ . \tag{4}$$

Let $P_{X|Y=y} = P_{X|Y=y,Z=z}$ for all y, z (or $P_{Z|Y=y} = P_{Z|Y=y,X=x}$ for all y, z, which is equivalent). Then we say that X, Y and Z form a Markov-chain, denoted by $X \leftrightarrow Y \leftrightarrow Z$. If $W \leftrightarrow XZ \leftrightarrow Y$, then

$$I(X;Y \mid ZW) \leq I(X;Y \mid Z) \ . \tag{5}$$

[6] The Bernoulli distribution with parameter $p \in [0,1]$ takes on the value 1 with probability p and 0 otherwise.

2 Impossibility Results

2.1 Model and Security Definition

We will consider the following model: a trusted third party holds random variables (U, V) with a joint distribution P_{UV} and sends U to the sender and V to the receiver. The sender receives an input bit $b \in \{0, 1\}$. In the commit phase, the players exchange messages in several rounds. Let all the messages exchanged be M, which is a randomized function of (U, V, b). In the open phase, the sender sends b together with a value D_1 to the receiver. The receiver then sends a message E_1 to the receiver, who replies with a message D_2 and so on. Let $N := (D_1, E_1, D_2, E_2, \ldots, E_{t-1}, D_t)$ be the total communication in the open phase. (We assume that the number of rounds in the open phase is upper bounded by a constant t. By padding the protocol with empty rounds we can thus assume without loss of generality that the protocol uses t rounds in every execution.) Finally, the receiver accepts or rejects, which we model by a randomized function $F(b, V, M, N)$ that outputs 1 for accept and 0 for reject. Let the distribution in the honest setting be $P_{UVMN|B=b}$. We define three parameters that quantify the security for the sender and the receiver, respectively, and the correctness of the protocol.

- ε-*correct*: $\Pr[F(b, V, M, N) = 1] \geq 1 - \varepsilon$.
- β-*hiding*: $\delta(P_{VM|B=0}, P_{VM|B=1}) \leq \beta$.
- γ-*binding*: For any $b \in \{0, 1\}$ and for any malicious sender that is honest in the commit phase on input b and tries to open $1 - b$, we have $\Pr[F(1 - b, V, M, N') = 1] \leq \gamma$, where N' is the communication between the malicious sender and the honest receiver in the open phase.

Note that the above security conditions are not sufficient to prove the security of a protocol[7], but any sensible security definition for commitments implies these conditions. Since we only use the definition to prove the non-existence of certain protocols, this makes our result stronger.

2.2 Lower Bound for Multiple Bit Commitments

In the following we prove a lower bound on the mutual information between the randomness of the sender and the randomness of the receiver in any bit commitment protocol. First, we show the following technical lemma.

Lemma 1. *If a protocol that implements bit commitment from distributed randomness (U, V) is γ-binding, ε-correct and β-hiding, then for $b \in \{0, 1\}$*

$$\mathrm{d}(1 - \varepsilon \,\|\, \gamma + \beta) \leq \sum_{i=1}^{t} \mathrm{I}(D_i; V \mid MD^{i-1}E^{i-1}, B = b). \tag{6}$$

[7] To prove the security of a protocol one had to consider for example a malicious sender in the commit phase.

Proof. Let $b \in \{0,1\}$ and $\bar{b} := 1 - b$. Assume that the sender in the commit phase honestly commits to b. If she honestly opens b in the open phase, the communication can be modeled by a channel $P_{DE|VM}$ (that may depend on b) and the resulting distribution is

$$P_{DEVM|B=b} = P_{DE|VM} P_{VM|B=b} \ ,$$

We have omitted U as it does not play a role in the following arguments. The correctness property implies that an honest receiver accepts values drawn from this distribution with probability at least $1 - \varepsilon$. Let the sender commit to $\bar{b}$ and then try to open b by sampling her messages according to the distributions $P_{D_1|M}$ and $P_{D_i|MD^{i-1}E^{i-1}}$ for $2 \leq i \leq t$. (Note that the sender does not know V and, therefore, chooses her messages independently of V.) The communication in the opening phase can be modeled by a channel

$$Q_{DE|VM} := P_{D_1|M} P_{E_1|VMD_1} \dots P_{D_t|MD^{t-1}E^{t-1}} \ .$$

The binding property implies that the receiver accepts values distributed according to $P_{VM|B=\bar{b}} Q_{DE|VM}$ with probability at most γ. $\delta(P_{VM|B=b}, P_{VM|B=\bar{b}}) \leq \beta$ implies that

$$\delta(P_{VM|B=b} Q_{DE|VM}, P_{VM|B=\bar{b}} Q_{DE|VM}) \leq \beta,$$

and hence values drawn from the distribution $P_{VM|B=b} Q_{DE|VM}$ are accepted with probability at most $\gamma + \beta$. Note that the bit indicating acceptance can also be modeled by a channel $P_{F|DEVM}$. Thus, we can apply the data-processing inequality (3) to bound $\mathrm{d}(1 - \varepsilon \parallel \gamma + \beta)$. Using the chain rule (1) and the non-negativity of the relative entropy, we have (we omit conditioning on $B = b$ in the following)

$$\begin{aligned}
\mathrm{d}(1 - \varepsilon \parallel \gamma + \beta) &\leq \mathrm{D}(P_{VM} P_{DE|VM} \parallel P_{VM} Q_{DE|VM}) \\
&= \mathrm{D}(P_{DE|VM} \parallel Q_{DE|VM}) \\
&= \sum_{i=1}^{t} \mathrm{D}(P_{D_i|VMD^{i-1}E^{i-1}} \parallel P_{D_i|MD^{i-1}E^{i-1}}) \\
&\qquad + \sum_{i=1}^{t-1} \mathrm{D}(P_{E_i|VMD^iE^{i-1}} \parallel P_{E_i|VMD^iE^{i-1}}) \\
&= \sum_{i=1}^{t} \mathrm{D}(P_{D_i|VMD^{i-1}E^{i-1}} \parallel P_{D_i|MD^{i-1}E^{i-1}}) \\
&= \sum_{i=1}^{t} \mathrm{I}(D_i; V \mid MD^{i-1}E^{i-1})
\end{aligned}$$

□

The following lemma follows easily from Theorem 2.1 in [31]. We will use it to bound the right-hand side of (6) in the following.

Lemma 2. *Let $\varepsilon = \beta = \gamma = 2^{-k}$. Then, for $k \geq 3$, we have*

$$\mathrm{d}(1-\varepsilon \parallel \gamma+\beta) \geq (k-2) \cdot \frac{2^{k-2}-2}{2^{k-2}-1}.$$

The following lemma generalizes the lower bounds on the size of the randomness for perfectly correct and perfectly hiding non-interactive schemes from [30] to arbitrary protocols. However, it also provides a more powerful result, namely a lower bound on the information that the communication in the open phase must reveal about the receiver's randomness V for any protocol that implements bit commitment from a shared distribution P_{UV}. The lower bound is essentially k if the error of the protocol is at most 2^{-k}. This stronger statement will allow us in the following to prove that there are no constant rate reductions of bit commitment to distributed randomness, the main result of this section.

Lemma 3. *Let $k \geq 3$. Then any 2^{-k}-secure bit commitment must have for $b \in \{0,1\}$*

$$\begin{aligned}&\mathrm{I}(N;V \mid M, B=b) - \mathrm{I}(N;V \mid UM, B=b)\\ &\quad = \mathrm{I}(U;V \mid M, B=b) - \mathrm{I}(U;V \mid MN, B=b) \geq (k-2) \cdot \frac{2^{k-2}-2}{2^{k-2}-1}\ .\end{aligned}$$

Proof. Again, we omit conditioning on $B = b$ in the following. Consider a protocol over t rounds in the open phase, i.e., the whole communication is $N = (D,E) = (D_1, E_1, \ldots, D_t)$. Since $D_i \leftrightarrow UMD^{i-1}E^{i-1} \leftrightarrow V$, we have $\mathrm{I}(D_i; V \mid UMD^{i-1}E^{i-1}) = 0$. Hence,

$$\mathrm{I}(NU;V \mid M) = \mathrm{I}(U;V \mid M) + \sum_{i=1}^{t-1} \mathrm{I}(E_i;V \mid UMD^iE^{i-1})\ .$$

Furthermore, from $E_i \leftrightarrow VMD^iE^{i-1} \leftrightarrow U$ and inequality (5) follows that for all i

$$\mathrm{I}(E_i;V \mid MD^iE^{i-1}) \geq \mathrm{I}(E_i;V \mid UMD^iE^{i-1})\ .$$

Hence, we have

$$\begin{aligned}\mathrm{I}(N;V \mid M) &= \sum_i \mathrm{I}(E_i;V \mid MD^iE^{i-1}) + \sum_i \mathrm{I}(D_i;V \mid MD^{i-1}E^{i-1})\\ &\geq \sum_i \mathrm{I}(E_i;V \mid UMD^iE^{i-1}) + \sum_i \mathrm{I}(D_i;V \mid MD^{i-1}E^{i-1})\end{aligned}$$

and

$$\begin{aligned}\mathrm{I}(U;V \mid MN) &= \mathrm{I}(NU;V \mid M) - \mathrm{I}(N;V \mid M)\\ &= \mathrm{I}(U;V \mid M) + \sum_i \mathrm{I}(E_i;V \mid UMD^iE^{i-1}) - \mathrm{I}(N;V \mid M)\\ &\leq \mathrm{I}(U;V \mid M) - \sum_i \mathrm{I}(D_i;V \mid MD^{i-1}E^{i-1})\ .\end{aligned}$$

The statement now follows from Lemma 1 and Lemma 2. □

Next, we consider implementations of n individual bit commitments. The sender gets input $b^n = (b_1, \ldots, b_n)$ and commits to all bits at the same time, which results in the overall distribution

$$P_{UVM|B^n=b^n} = P_{UV} P_{M|UV,B^n=b^n} \ .$$

after the commit phase. To reveal the ith bit, the sender and the receiver interact resulting in the transcript N_i. The following theorem says that the mutual information between the sender's randomness U and the receiver's randomness V must be almost kn to implement n bit commitments with an error of at most 2^{-k}. The proof uses Lemma 3 to lower bound the information that the sender must reveal about V for every bit that he opens.

Theorem 1. *Let $k \geq 3$. Then any 2^{-k}-secure protocol that implements n bit commitments from randomness (U, V) must have for all $b^n \in \{0,1\}^n$*

$$\mathrm{I}(U;V) \geq \mathrm{I}(U;V \mid M, B = b^n) \geq n(k-2) \cdot \frac{2^{k-2}-2}{2^{k-2}-1} \ .$$

Proof. Let $\hat{i} \in [n]$. We first construct a commitment to a single bit, which will allow us to apply the bound from Lemma 3. This bit commitment is defined as follows: to commit to the bit b, the players execute the commit phase on input b^n, which is equal to the input bit b on position $\hat{i}$ and equal to the constant $\hat{b}^n \in \{0,1\}^n$ on all other positions. Additionally, (still as part of the commit phase), the sender opens the first $\hat{i}-1$ commitments, which means that the messages $N^{\hat{i}-1}$ get exchanged. To open the commitment, the sender opens bit $\hat{i}$. This bit commitment scheme has at least the same security as the original commitment. Thus, Lemma 3 implies that (we omit conditioning on $B = \hat{b}^n$ in the following)

$$\mathrm{I}(U;V \mid MN^{\hat{i}}) \leq \mathrm{I}(U;V \mid MN^{\hat{i}-1}) - (k-2) \cdot \frac{2^{k-2}-2}{2^{k-2}-1}. \tag{7}$$

Since this holds for all $\hat{i}$, we can apply (7) repeatedly to get

$$\begin{aligned} 0 &\leq \mathrm{I}(U;V \mid MN^n) \\ &\leq \mathrm{I}(U;V \mid MN^{n-1}) - (k-2) \cdot \frac{2^{k-2}-2}{2^{k-2}-1} \\ &\leq \mathrm{I}(U;V \mid M) - n(k-2) \cdot \frac{2^{k-2}-2}{2^{k-2}-1} \end{aligned}$$

By induction over all rounds of the commit protocol using (5) (see, for example, [37] for a detailed proof) it follows that

$$I(U;V \mid M) \leq I(U;V) \ .$$

This implies the statement. □

It is possible to securely implement 1-out-of-2 bit oblivious transfer $\binom{2}{1}$-OT^1 from randomness distributed according to P_{UV} with $\mathrm{I}(U;V) = 1$ [3, 1]. A binary symmetric noisy channel ((p)-BSNC) with crossover probability p can be implemented from randomness distributed according to P_{UV} with $\mathrm{I}(U;V) = 1 - h(p)$. Together with these reductions, Theorem 1 implies that (almost) kn instances of $\binom{2}{1}$-OT^1 or $kn/(1-h(p))$ instances of a (p)-BSNC are needed to implement n bit commitments with an error of at most 2^{-k}.

There exists a universally composable protocol[8] that implements bit commitment from $2k$ instances of $\binom{2}{1}$-OT^1 with an error of at most 2^{-k}. Thus, n bit commitments can be implemented from $2n(k + \log(n))$ instances of $\binom{2}{1}$-OT^1 with an error of at most $n \cdot 2^{-(k+\log(n))} = 2^{-k}$ using n parallel instances of this protocol. Theorem 1 shows that this is optimal up to a factor of 4 if $k \geq \log(n)$.

2.3 Lower Bounds for Multiple String Commitments

A *string commitment* is a generalization of bit commitment where the sender may commit to a bit-string of length $\ell \geq 1$. It is weaker than ℓ instances of bit commitment because the sender has to reveal all bits simultaneously. In [38] a lower bound on the conditional entropy of the sender's randomness U given the receiver's randomness V for any string commitment protocol from randomness (U, V) has been shown. This bound essentially says that $\mathrm{H}(U \mid V)$ must be greater than or equal to ℓ to implement a string commitment of length ℓ. The following lemma provides a similar bound for the security definition considered here. (The proof can be found in the full version of this paper [33].)

Lemma 4. *If any protocol implements an ℓ-bit string commitment from randomness (U, V) is ε-correct, $\beta-$hiding and γ-binding, then*

$$\mathrm{H}(U \mid V) \geq (1 - \varepsilon - \beta - \gamma)\ell - h(\beta) - h(\varepsilon + \gamma).$$

Together with the bound of Theorem 1, we obtain the following lower bound on the randomness of the sender in any bit commitment protocol.

Corollary 1. *Let $k \geq 3$. For any protocol that implements n individual ℓ-bit string commitments from randomness (U, V) with an error of at most 2^{-k}*

$$\mathrm{H}(U) \geq n(k + \ell - 2) \cdot \frac{2^{k-2} - 2}{2^{k-2} - 1} - 3 \cdot 2^{-k} \cdot n\ell - 3h(2^{-k}).$$

Proof. Using Lemma 4 and Theorem 1, we get

$$\begin{aligned}
\mathrm{H}(U) &= \mathrm{I}(U;V) + \mathrm{H}(U \mid V) \\
&\geq n(k-2) \cdot \frac{2^{k-2} - 2}{2^{k-2} - 1} + (1 - 3 \cdot 2^{-k})n\ell - h(2^{-k}) - h(2^{-k+1}) \\
&\geq n(k + \ell - 2) \cdot \frac{2^{k-2} - 2}{2^{k-2} - 1} - 3 \cdot 2^{-k} \cdot n\ell - 3h(2^{-k}).
\end{aligned}$$

□

[8] See for example Claim 33 in the full version of [8].

In [5] it has been shown that any non-interactive perfectly hiding and perfectly correct bit commitment protocol from distributed randomness P_{UV} is at most $(2^{-H(V|U)})$-binding. This result implies stronger bounds than Theorem 1 and Lemma 4 for certain reductions. The following lemma provides a lower bound on the uncertainty of the sender about the receiver's randomness for any bit commitment protocol. This lower bound is essentially equal to k if the protocol is 2^{-k}-secure and implies, in particular, the result from [5].

Lemma 5. *If a protocol that implements bit commitment from randomness (U, V) is γ-binding, ε-correct and β-hiding, then*

$$\mathrm{d}(1-\beta-\varepsilon \parallel \gamma) \leq \mathrm{H}(V \mid UM) \leq \mathrm{H}(V \mid U).$$

where M is the whole communication in the commit phase. If $\beta = \gamma = \varepsilon = 2^{-k}$, then

$$\mathrm{H}(V \mid U) \geq (k-1) \cdot \frac{2^{k-1}-4}{2^{k-1}-1}. \tag{8}$$

Proof. We have $\delta(P_{VM|B=b}, P_{VM|B=\bar{b}}) \leq \beta$. This implies that the distribution $P_{U|VM,B=\bar{b}} P_{VM|B=b}$ is β-close to $P_{UVM|B=\bar{b}}$. Thus, when the sender honestly opens $\bar{b}$ starting from values distributed according $P_{U|VM,B=\bar{b}} P_{VM|B=b}$, the receiver accepts the resulting values with probability at least $1-\beta-\varepsilon$. We consider the following attack: the sender honestly commits to b, generates v' by applying $P_{V|UM,B=b}$ and then generates u by applying the channel $P_{U|VM,B=\bar{b}}$ to (v', m). When the sender now tries to open $\bar{b}$, the binding property guarantees that the receiver accepts the resulting values with probability at most γ. Thus, we can apply the data-processing inequality (3) to bound $\mathrm{d}(1-\beta-\varepsilon \parallel \gamma)$. Let V' be a copy of V, i.e., a random variable with distribution $P_{VV'}(v, v) = P_V(v)$. Using the chain rule (2), we have

$$\begin{aligned}
\mathrm{d}(1-\beta-\varepsilon \parallel \gamma) &\leq \mathrm{D}(P_{VV'|UM,B=b} P_{UM|B=b} \parallel P_{V|UM,B=b} P_{V|UM,B=b} P_{UM|B=b}) \\
&\leq \mathrm{D}(P_{VV'|UM,B=b} \parallel P_{V|UM,B=b} P_{V|UM,B=b}) \\
&= \mathrm{H}(V \mid UM, B=b) \\
&\leq \mathrm{H}(V \mid U).
\end{aligned}$$

Using Lemma 2 this implies inequality (8). □

Consider a protocol that implements n bit commitment with security of 2^{-k} from n' instances of $\binom{2}{1}$-$\mathsf{OT}^{\ell'}$. Since $\binom{2}{1}$-$\mathsf{OT}^{\ell'}$ can be reduced to a shared distribution P_{UV} with $H(V|U) = 1$, Lemma 5 implies that $n' \geq (k-1) \cdot \frac{2^{k-1}-4}{2^{k-1}-1}$, i.e., one needs, independently of ℓ', almost k instances of OT.

Together with Theorem 1 and Lemma 4, this implies the following lower bound on the number of instances of OT needed to implement multiple string commitments, which demonstrates that all three lower bounds can be meaningful in this scenario.

Corollary 2. *Let $k \geq 3$. For any protocol that implements n individual ℓ-bit string commitments with an error of at most 2^{-k} from n' instances of $\binom{2}{1}$-$OT^{\ell'}$*

$$n' \geq \max\left(\frac{\ell n}{\ell'}(1 - 3 \cdot 2^{-k}) - \frac{3h(2^{-k})}{\ell'}, \frac{(k-2)n}{\ell'} \cdot \frac{2^{k-2}-2}{2^{k-2}-1}, (k-1)\frac{2^{k-1}-4}{2^{k-1}-1}\right).$$

3 Commitments with restricted openings

In this section, we will present a protocol that implements commitments with restricted openings from several instances of string commitment. We will use the Universal Composability model [8], and assume that the reader is familiar with it. In our proof, we will only consider static adversaries. For simplicity, we omit session IDs and players IDs.

String Commitment is a functionality that allows the sender to commit to a string of n bits, and to reveal the whole string later to the receiver. The receiver does not get to know the string before it is opened, and the sender cannot change the string once he has sent it.

Definition 1 (String-Commitment). *The functionality $\mathcal{F}_{SCOM}^{n}$ behaves as follows:*

- *Upon input (`commit`, b) with $b \in \{0,1\}^n$ from the sender: check that commit has not been sent yet. If so, send `committed` to the receiver and store b. Otherwise, ignore the message.*
- *Upon input `openall` from the sender: check if there has been a commit message before, and the commitment has not been opened yet. If so, send (`openall`, b) to the receiver and ignore the message otherwise.*

Note that given $\mathcal{F}_{\mathtt{SCOM}}^{n}$, it is possible to commit to individual bits at different times: the sender simply commits to a random string $b' = (b'_1, \ldots, b'_n)$, and whenever he wants to commit to a bit b_i for $i \in [n]$, he sends $b_i \oplus b'_i$ to the receiver. On the other hand, it is not possible to open bits at different times using $\mathcal{F}_{\mathtt{SCOM}}^{n}$.

Bit commitment is a string commitment of length 1, i.e., $\mathcal{F}_{\mathtt{BCOM}} := \mathcal{F}_{\mathtt{SCOM}}^{1}$. We denote n independent bit commitments by $(\mathcal{F}_{\mathtt{BCOM}})^n$. Since $(\mathcal{F}_{\mathtt{BCOM}})^n$ does allow bits to be opened at different times, it is strictly stronger than $\mathcal{F}_{\mathtt{SCOM}}^{n}$. However, as we have seen in the last section, $(\mathcal{F}_{\mathtt{BCOM}})^n$ is also quite expensive to implement in terms of resources needed. Therefore, we define a primitive that is somewhere between these two: *commitments with restricted openings* allow a sender to commit to n bits, but then he may only open r individual bits of his choice one by one. To open more than r bits, he has to open the remaining bits all at once.

Definition 2 (Commitments with restricted openings). *The functionality $\mathcal{F}_{RCOM}^{n,r}$ behaves as follows:*

– *Upon input (*`commit`*, b) with* $b \in \{0,1\}^n$ *from the sender: check that commit has not been sent yet. If so, send* `committed` *to the receiver and store b. Otherwise, ignore the message.*
– *Upon input (*`open`*,i) with* $i \in [n]$ *from the sender: check that there has been a commit message before, and that i has not been opened yet. Also check that the number of opened values so far is smaller than r. If so, send (*`open`*, i,* b_i*) to the receiver and ignore the message otherwise.*
– *Upon input* `openall` *from the sender: check if there has been a commit message before, and no* `openall` *message has been received yet from the sender. If so, send (*`openall`*, b) to the receiver and ignore the message otherwise.*

For $r = 0$ and $r = n$, commitment with restricted openings are equivalent to string commitments and individual bit commitments, respectively: $\mathcal{F}_{\mathtt{SCOM}}^{n} = \mathcal{F}_{\mathtt{RCOM}}^{n,0}$ and $(\mathcal{F}_{\mathtt{BCOM}})^n \equiv \mathcal{F}_{\mathtt{RCOM}}^{n,n}$.

Our protocol makes use of *cover-free families* [27, 18, 35, 6], which are a generalization of *Sperner sets* [34]. Cover-free families are also known as *superimposed codes* and require that no set is covered by the union of r other sets.

Definition 3. *Let* $\mathcal{X}$ *be a set of n elements and let* $\mathcal{B}$ *be a set of subsets of* $\mathcal{X}$*, then* $(\mathcal{X}, \mathcal{B})$ *is a r-cover-free family* $r-\mathrm{CFF}(\mathcal{X}, \mathcal{B})$ *if for any r sets* $B_{i_1}, \ldots B_{i_r} \in \mathcal{B}$*, and any other* $B \in \mathcal{B}$*, it holds that*

$$B \not\subseteq \bigcup_{j=1}^{r} B_{i_j} \ .$$

Example 1. All subsets of $[n]$ of size s form a cover-free family for $r = 1$, because there is no subset that completely covers any other subset.

Here is a simple example of a cover-free family for $r > 1$ given in [18].

Example 2 ([18]). Let q be a prime power, and $d, r \in \mathbb{N}$ such that $rd < q$. Let $\mathcal{X} = \mathcal{Y} \times GF(q)$, where $\mathcal{Y} \subseteq GF(q)$ and $|\mathcal{Y}| = rd+1$. An element B in the family $\mathcal{B}$ is constructed from a polynomial $p(y) := a_0 + y \cdot a_1 + ... + y^d \cdot a_d$ of degree d where $a_i \in \mathrm{GF}(q)$ by $B := \{(y, p(y)) : y \in \mathcal{Y})\}$. Two polynomials of degree d intersect at most d times. Therefore, any union of r elements $B_1, \ldots B_r$ intersects any other element B at most $rd < |\mathcal{Y}|$ times, and therefore cannot cover B. $(\mathcal{X}, \mathcal{B})$ is therefore a r-cover-free family with $|\mathcal{X}| = (rd+1)q$ and $|\mathcal{B}| = q^{d+1}$.

We now give a protocol that implements $\mathcal{F}_{\mathtt{RCOM}}^{N,r}$ from n instances of $\mathcal{F}_{\mathtt{SCOM}}^{N}$ using a $r-\mathrm{CFF}(\mathcal{X}, \mathcal{B})$, where $\mathcal{X} = \{1, \ldots, n\}$ and $\mathcal{B} = \{B_1, B_2, ..., B_N\}$.

Protocol 1:

- When the sender receives (`commit`, b), he chooses n uniformly chosen strings $c_1, \ldots c_n \in \{0,1\}^N$, with the restriction that for all $i \in [N]$ we have
$$\bigoplus_{j \in B_i} c_{j,i} = b_i \ .$$
For $j \in [n]$, the sender sends (`commit`, c_j) to the jth instances of $\mathcal{F}_{\texttt{SCOM}}^N$. After that he ignores all messages (`commit`, b').
- When the receiver has received `committed` from all instances of $\mathcal{F}_{\texttt{SCOM}}^N$, he outputs `committed`.
- For the first r times when the sender receives `(open,`i`)`, he sends `(open,`i`)` to the receiver and `openall` to all instances of $\mathcal{F}_{\texttt{SCOM}}^N$ in B_i, if they have not been opened yet. After that, he ignores all messages `(open,`i`)`.
- For the first r times when the receiver receives `(open,`i`)` from the sender and (`open`, c_j) from all instances $\mathcal{F}_{\texttt{SCOM}}^N$ in B_i, he outputs (`open`, $\bigoplus_{j \in B_i} c_{j,i}$). After that, he ignores these messages.
- When the sender receives `openall`, he sends `openall` to the receiver and to all instances of $\mathcal{F}_{\texttt{SCOM}}^N$. After that, he ignores all `openall` messages.
- When the receiver receives `openall` from the sender and (`open`, c_j) from all instances of $\mathcal{F}_{\texttt{SCOM}}^N$, he outputs (`openall`, $(b'_1, \ldots, b'_N)$), where $b'_i := \oplus_{j \in B_i} c_{j,i}$. After that, he ignores all messages `openall`.

Theorem 2. *Given an* $r-\mathrm{CFF}(\mathcal{X}, \mathcal{B})$ *where* $|\mathcal{X}| = n$ *and* $|\mathcal{B}| = N$*, Protocol* 1 *UC-implements* $\mathcal{F}_{\texttt{RCOM}}^{N,r}$ *from* n *instances of* $\mathcal{F}_{\texttt{SCOM}}^N$*.*

Proof. It is easy to verify that the protocol is correct if the two players are honest.

Corrupted sender. First, we consider the case where the comitter is corrupted. He may send messages (`commit`, c_j) or `openall` to the instances of $\mathcal{F}_{\texttt{SCOM}}^N$, and message `(open,`$i$`)` or `openall` to the receiver.

Our simulator simulates the adversary, and records all messages sent out by the adversary. After receiving all messages (`commit`, c_j) to the instances of $\mathcal{F}_{\texttt{SCOM}}^N$, he calculates $b_i := \oplus_{j \in B_i} c_{j,i}$ for all i and sends (`commit`, $(b_1, \ldots, b_N)$) to $\mathcal{F}_{\texttt{RCOM}}^{N,r}$. After receiving `(open,`$i$`)` and all messages `openall` sent to the instances of $\mathcal{F}_{\texttt{SCOM}}^N$ in B_i, he sends `(open,`i`)` to $\mathcal{F}_{\texttt{RCOM}}^{N,r}$. After receiving `openall` sent to the receiver and all instances $\mathcal{F}_{\texttt{SCOM}}^N$, he sends `openall` to $\mathcal{F}_{\texttt{RCOM}}^{N,r}$. It is not difficult to verify that our simulation is perfect, and we get REAL$\equiv$ IDEAL.

Corrupted receiver. Let the receiver be corrupted by the adversary. He receives `committed` and (`open`,c_j) messages from the instances of $\mathcal{F}_{\texttt{SCOM}}^N$, and messages `(open,`$i$`)` and `openall` from the sender.

Our simulator simulates the adversary, and interacts with $\mathcal{F}_{\texttt{RCOM}}^{N,r}$ and the adversary. After receiving the `committed` message from $\mathcal{F}_{\texttt{RCOM}}^{N,r}$, it sends `committed` from all $\mathcal{F}_{\texttt{SCOM}}^{N}$ to the adversary. After receiving message `(open,`i`,`b_i`)` from $\mathcal{F}_{\texttt{RCOM}}^{N,r}$, he first sends `(open,`$i$`)` to the adversary. Then for all instances of $\mathcal{F}_{\texttt{SCOM}}^{N}$ in B_i which have not been opened yet, he chooses strings c_j uniformly at random, with the restriction that $\oplus_{j\in B_i} c_{j,i} = b_i$, and sends `(open,`$c_j$`)` from the jth instance of $\mathcal{F}_{\texttt{SCOM}}^{N}$ to the adversary. After receiving message `(openall,`b`)` from $\mathcal{F}_{\texttt{RCOM}}^{N,r}$, he first sends `openall` to the adversary. Then for all instances of $\mathcal{F}_{\texttt{SCOM}}^{N}$ which have not been opened yet, he chooses the strings c_j uniformly at random, with the restriction that $\oplus_{j\in B_i} c_{j,i} = b_i$, and sends `(open,`$c_j$`)` from the jth instance of $\mathcal{F}_{\texttt{SCOM}}^{N}$ to the adversary.

To show that this simulation in the ideal setting is identical to the real setting, we have to show that they are identical after each step. It is easy to see that this is the case before anything has been opened, and after `openall` has been executed.

$\mathcal{F}_{\texttt{RCOM}}^{N,r}$ allows the sender to open at most r values. Assume that $s \leq r$ have been opened so far. Since $\mathcal{B}$ is a $r-\mathrm{CFF}(\mathcal{X},\mathcal{B})$, there is at least one instance of $\mathcal{F}_{\texttt{SCOM}}^{N}$ in B_i for all the remaining $i \in [N]$ that has not been opened yet. Since the ith bit of that string is uniform and all the ith bits of the strings in B_i add up to b_i, the bits at the ith position of all the opened strings are uniform and independent of each other and of the bit b_i. Therefore, the simulated values c_j sent to the adversary have the same distribution in the real and in the ideal setting. The simulation is again perfect, and we get REAL$\equiv$ IDEAL. □

Note that in each instance of $\mathcal{F}_{\texttt{SCOM}}^{N}$ in Protocol 1, only a subset of the bits are actually used. Since they are at fixed positions and both players know where they are, they can be removed without changing the properties of the protocol. If we use the cover-free family from Example 1, the length of the string commitments used can be reduced to Ns/n, and we get the following corollary.

Corollary 3. *For any $n \geq s \geq 1$ and $N = \binom{n}{s}$ there exists a protocol that UC-implements $\mathcal{F}_{RCOM}^{N,1}$ from $\left(\mathcal{F}_{SCOM}^{Ns/n}\right)^n$.*

The protocol is optimal in the length of the strings up to a factor s; otherwise it would be possible to implement a string commitment of length bigger than $n \cdot \ell$ from n instances of string commitment of length ℓ, which is not possible. Thus, we can build $N = n(n-1)/2$ bit commitments (choosing $s = 2$), from which one can be opened, from n string commitments of length $n-1$. When choosing $s = n/2$, we obtain an exponential number of committed bits from n strings, since $N = \binom{n}{n/2} > 2^{n/2}$.

If we use the cover-free family of Example 2, then the size of the commitments can be reduced by a factor of q because we can let all the bit commitments which have different values a_0 but the same values $a_1, ..., a_d$ share the same position in the string commitments. We get the following corollary.

Corollary 4. *Let q be a prime power, $d < q$ and $N := q^{d+1}$. There exists a protocol that UC-implements $\mathcal{F}_{RCOM}^{N,r}$ from $(rd+1)q$ instances of $\mathcal{F}_{SCOM}^{N/q}$.*

This is optimal in the length of the strings up to a factor $rd + 1$; otherwise it would again be possible to implement a string commitment of length bigger than $n \cdot \ell$ from n instances of string commitment of length ℓ, which is not possible. Choosing $d = 1$, we get $N = q^2$ and $n = (r + 1)q$. Thus, there exists a protocol that uses $(r + 1)q$ string commitments of length q and implements q^2 bit commitments from which r can be opened.

To obtain an exponential number of bit commitments from n string commitments, we can use Corollary 1 in [6] which gives an explicit construction of a $t-\mathrm{CFF}(\mathcal{X}, \mathcal{B})$ where $|\mathcal{X}| < 24t^2 \log(|\mathcal{B}| + 2)$. Hence, we get the following result.

Corollary 5. *There exists a protocol that from $\mathcal{F}_{RCOM}^{N,r}$ from $24r^2 \log(N + 2)$ instances of $\mathcal{F}_{SCOM}^{N}$.*

This is close to the optimal efficiency we can expect from Protocol 1, as it has been shown in Theorem 1.1 in [35] that $t-\mathrm{CFF}(\mathcal{X}, \mathcal{B})$ must have

$$|\mathcal{X}| \geq c \cdot \frac{t^2}{\log t} \log |\mathcal{B}| \ ,$$

for a constant c.

Our protocols can be generalized in a simple way as follows: let $\mathcal{F}_{\mathtt{RCOM}}^{N,r,c}$ be the same functionality as $\mathcal{F}_{\mathtt{RCOM}}^{N,r}$ except that every bit is replaced by a block of size c. The sender can open up to r blocks, or all N blocks at the same time. It is not difficult to see that if Protocol 1 implements $\mathcal{F}_{\mathtt{RCOM}}^{N,r}$ from n instances of $\mathcal{F}_{\mathtt{SCOM}}^{\ell}$, then it can be transformed into a protocol that implements $\mathcal{F}_{\mathtt{RCOM}}^{N,r,c}$ from n instances of $\mathcal{F}_{\mathtt{SCOM}}^{\ell c}$.

3.1 Commitments from Noisy Channels at a Constant Rate

From Corollary 4 with $d = 1$ in combination with the string commitment protocol presented in [38], we get the following corollary.

Corollary 6. *For any constant r, there exists a protocol that implements $\mathcal{F}_{RCOM}^{n,r}$ using only $O(n)$ noisy channels.*

This is optimal up to a constant factor.

4 Conclusions

In this work we have shown a strong lower bound for reductions of multiple bit commitments to other information theoretic primitives, such as oblivious transfer or noisy channels. Our bound shows that every single bit commitment needs at least $\Omega(k)$ instances of the underlying primitive. This makes bit commitments often much more costly to implement than oblivious transfer, for example. It would be interesting to see whether these results can be generalized to other functionalities.

We have presented a protocol that implements bit commitments more efficiently, when the number of bits that can be opened is restricted. Our protocol implements commitments with restricted openings from string commitments. We think that for some resources more efficient protocols might be possible by implementing them directly, instead of using string commitments as a building block.

Acknowledgements

AT is supported by Canada NSERC. SW is supported from the Swiss National Science Foundation and an ETHIIRA grant of ETH's research commission. JW is supported by the Canada-France NSERC-ANR project FREQUENCY.

References

1. D. Beaver. Precomputing oblivious transfer. In *Advances in Cryptology — EUROCRYPT '95*, volume 963 of *Lecture Notes in Computer Science*, pages 97–109. Springer-Verlag, 1995.
2. Amos Beimel and Tal Malkin. A quantitative approach to reductions in secure computation. In *Theory of Cryptography Conference — TCC '04*, pages 238–257, 2004.
3. C. H. Bennett, G. Brassard, C. Crépeau, and H. Skubiszewska. Practical quantum oblivious transfer. In *Advances in Cryptology — CRYPTO '91*, volume 576 of *Lecture Notes in Computer Science*, pages 351–366. Springer, 1992.
4. M. Blum. Coin flipping by telephone a protocol for solving impossible problems. *SIGACT News*, 15(1):23–27, 1983.
5. C. Blundo, B. Masucci, D. R. Stinson, and R. Wei. Constructions and bounds for unconditionally secure non-interactive commitment schemes. *Des. Codes Cryptography*, 26:97–110, June 2002.
6. A. De Bonis and U. Vaccaro. Constructions of generalized superimposed codes with applications to group testing and conflict resolution in multiple access channels. *Theor. Comput. Sci.*, 306:223–243, September 2003.
7. G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *J. Comput. Syst. Sci.*, 37:156–189, October 1988.
8. R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *Proceedings of the 42th Annual IEEE Symposium on Foundations of Computer Science (FOCS '01)*, pages 136–145, 2001. Updated Version at http://eprint.iacr.org/2000/067.
9. T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, New York, USA, 1991.
10. C. Crépeau. Equivalence between two flavours of oblivious transfers. In *Advances in Cryptology — EUROCRYPT 1987*, Lecture Notes in Computer Science, pages 350–354. Springer-Verlag, 1988.
11. C. Crépeau. Efficient cryptographic protocols based on noisy channels. In *Advances in Cryptology — CRYPTO '97*, volume 1233 of *Lecture Notes in Computer Science*, pages 306–317. Springer-Verlag, 1997.

12. C. Crépeau and J. Kilian. Achieving oblivious transfer using weakened security assumptions (extended abstract). In *Proceedings of the 29th Annual IEEE Symposium on Foundations of Computer Science (FOCS '88)*, pages 42–52, 1988.
13. C. Crépeau, K. Morozov, and S. Wolf. Efficient unconditional oblivious transfer from almost any noisy channel. In *Proceedings of Fourth Conference on Security in Communication Networks (SCN)*, volume 3352 of *Lecture Notes in Computer Science*, pages 47–59. Springer-Verlag, 2004.
14. C. Crépeau, J. van de Graaf, and A. Tapp. Committed oblivious transfer and private multi-party computation. In *Advances in Cryptology — CRYPTO '95*, pages 110–123. Springer-Verlag, 1995.
15. I. Damgård, S. Fehr, K. Morozov, and L. Salvail. Unfair noisy channels and oblivious transfer. In *Theory of Cryptography Conference — TCC '04*, volume 2951 of *Lecture Notes in Computer Science*, pages 355–373. Springer-Verlag, 2004.
16. I. Damgård, J. Kilian, and L. Salvail. On the (im)possibility of basing oblivious transfer and bit commitment on weakened security assumptions. In *Advances in Cryptology — EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 56–73. Springer-Verlag, 1999.
17. Y. Dodis and S. Micali. Lower bounds for oblivious transfer reductions. In *Advances in Cryptology — EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 42–55. Springer-Verlag, 1999.
18. P. Erdős, P. Frankl, and Z. Füredi. Families of finite sets in which no set is covered by the union of r others. *Israel Journal of Mathematics*, 51(1–2):79–89, 1985.
19. S. Even, O. Goldreich, and A. Lempel. A randomized protocol for signing contracts. *Commun. ACM*, 28(6):637–647, 1985.
20. O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems. *J. ACM*, 38(3):690–728, 1991.
21. S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof-systems. In *Proceedings of the 17th Annual ACM Symposium on Theory of Computing (STOC '85)*, pages 291–304. ACM Press, 1985.
22. Te Sun Han and Kingo Kobayashi. *Mathematics of Information and Coding*. American Mathematical Society, Boston, MA, USA, 2001.
23. D. Harnik, Y. Ishai, E. Kushilevitz, and J. B. Nielsen. OT-combiners via secure computation. In *Theory of Cryptography Conference — TCC '08*, 2008.
24. Y. Ishai, E. Kushilevitz, R. Ostrovsky, M. Prabhakaran, A. Sahai, and J. Wullschleger. Constant-rate oblivious transfer from noisy channels. In *CRYPTO*, 2011.
25. Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai. Extracting correlations. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS '09)*, pages 261–270, 2009.
26. Y. Ishai, M. Prabhakaran, and A. Sahai. Founding cryptography on oblivious transfer — efficiently. In *Advances in Cryptology — CRYPTO '08*, pages 572–591. Springer-Verlag, 2008.
27. W. Kautz and R. Singleton. Nonrandom binary superimposed codes. *IEEE Trans. on Information Theory*, 10(4):363–377, 1964.
28. J. Kilian. More general completeness theorems for secure two-party computation. In *Proceedings of the 32th Annual ACM Symposium on Theory of Computing (STOC '00)*, pages 316–324. ACM Press, 2000.
29. J. Kilian, S. Micali, and R. Ostrovsky. Minimum resource zero-knowledge proofs. In *Proceedings of the 30th Annual IEEE Symposium on Foundations of Computer Science (FOCS '89)*, pages 474–479. IEEE, 1989.

30. A. Nascimento, A. Otsuka, H. Imai, and Jöern Müller-Quade. Unconditionally secure homomorphic pre-distributed commitments. In Marc Fossorier, Tom Hholdt, and Alain Poli, editors, *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, volume 2643 of *Lecture Notes in Computer Science*, pages 604–604. Springer-Verlag, 2003.
31. E. Ordentlich and M.J. Weinberger. A distribution dependent refinement of pinsker's inequality. *Information Theory, IEEE Transactions on*, 51(5):1836 – 1840, May 2005.
32. M. O. Rabin. How to exchange secrets by oblivious transfer. Technical Report TR-81, Harvard Aiken Computation Laboratory, 1981.
33. S. Ranellucci, A. Tapp, S. Winkler, and J. Wullschleger. On the efficiency of bit commitment reductions. Cryptology ePrint Archive, Report 2011/324, 2011.
34. J. Sperner. Ein Satz über Untermengen einer endlichen Menge. *Math. Z.*, 27:544–548, 1928.
35. D. R. Stinson, R. Wei, and L. Zhu. Some new bounds for cover-free families. *J. Combin. Theory A*, 90:224–234, 1999.
36. S. Wiesner. Conjugate coding. *SIGACT News*, 15(1):78–88, 1983.
37. S. Winkler and J. Wullschleger. On the efficiency of classical and quantum oblivious transfer reductions. In *Advances in Cryptology — CRYPTO '10*, Lecture Notes in Computer Science. Springer-Verlag, 2010.
38. A. Winter, A. C. A. Nascimento, and H. Imai. Commitment capacity of discrete memoryless channels. In *IMA Int. Conf.*, pages 35–51, 2003.
39. S. Wolf and J. Wullschleger. New monotones and lower bounds in unconditional two-party computation. In *Advances in Cryptology — CRYPTO '05*, volume 3621 of *Lecture Notes in Computer Science*, pages 467–477, 2005.
40. J. Wullschleger. Oblivious-transfer amplification. In *Advances in Cryptology — EUROCRYPT '07*, Lecture Notes in Computer Science. Springer-Verlag, 2007.
41. J. Wullschleger. Oblivious transfer from weak noisy channels. In *Theory of Cryptography Conference — TCC '09*, 2009.
42. 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.