BRICS

Basic Research in Computer Science

Statistical Secrecy and Multi-Bit Commitments

Ivan B. Damgård
Torben P. Pedersen
Birgit Pfitzmann

BRICS Report Series RS-96-45

ISSN 0909-0878 November 1996

See back inner page for a list of recent publications in the BRICS Report Series. Copies may be obtained by contacting:

BRICS
Department of Computer Science
University of Aarhus
Ny Munkegade, building 540
DK - 8000 Aarhus C
Denmark

Telephone: +45 8942 3360
Telefax: +45 8942 3255
Internet: BRICS@brics.dk

BRICS publications are in general accessible through World Wide Web and anonymous FTP:

`http://www.brics.dk/`
`ftp://ftp.brics.dk/pub/BRICS`
This document in subdirectory `RS/96/45/`

Statistical Secrecy and Multi-Bit Commitments[*]

Ivan B. Damgård[†] Torben P. Pedersen[‡]
Birgit Pfitzmann[§]

Version Nov. 25, 1996

Abstract

We present and compare definitions of the notion of "statistically hiding" protocols, and we propose a novel statistically hiding commitment scheme. Informally, a protocol statistically hides a secret if a computationally unlimited adversary who conducts the protocol with the owner of the secret learns almost nothing about it. One definition is based on the L_1-norm distance between probability distributions, the other on information theory. We prove that the two definitions are essentially equivalent. For completeness, we also show that statistical counterparts of definitions of computational secrecy are essentially equivalent to our main definitions.

Commitment schemes are an important cryptologic primitive. Their purpose is to commit one party to a certain value, while hiding this

[*] A preliminary sketch of one result of this paper, the multi-bit commitment scheme, was presented in Section 4 of a paper by the same authors at Crypto '93.

[†] Aarhus University, BRICS (Basic Research in Computer Science, Center of the Danish National Research Foundation), Matematisk Institut, Ny Munkegade, DK-8000 Aarhus C, Denmark. e-mail: ivan@daimi.aau.dk.

[‡] Aarhus University, Matematisk Institut, Ny Munkegade, DK-8000 Aarhus C, Denmark. e-mail: tppedersen@daimi.aau.dk.

[§] Universität Hildesheim, Institut für Informatik, Marienburger Platz 22, D-31141 Hildesheim, Germany. e-mail: pfitzb@informatik.uni-hildesheim.de; future: pfitzb@ls6.informatik.uni-dortmund.de.

value from the other party until some later time. We present a statistically hiding commitment scheme allowing commitment to many bits. The commitment and reveal protocols of this scheme are constant round, and the size of a commitment is independent of the number of bits committed to. This also holds for the total communication complexity, except of course for the bits needed to send the secret when it is revealed. The proof of the hiding property exploits the equivalence of the two definitions.

Index terms — Cryptology, Shannon theory, unconditional security, statistically hiding, multi-bit commitment, similarity of ensembles of distributions, zero-knowledge, protocols.

1 Introduction

Suppose party A conducts a protocol with party B while using some secret or partly secret input x. How does one state precisely that even if B is unlimited and behaves arbitrarily, B learns almost nothing about x that he did not know before?

This question is of interest in cryptologic protocols. In particular, we consider commitment schemes. They consist of two protocols. In the first protocol, A commits to a value x, while keeping x secret from B. In the second protocol, which can take place much later, A releases x to B. Commitment schemes play an important role in designing other cryptologic protocols. For instance, a folklore protocol for common coin flipping is constructed as follows: First, A flips a secret coin c_1 and commits to it. Next, B flips a coin c_2 and publishes it. Finally, A reveals c_1, and $c_1 \oplus c_2$ is taken as the common coin that both A and B trust to be random. Commitment protocols are also important in constructing general perfect or statistical zero-knowledge protocols, see [23] or [6].

Generally, three types of secrecy are distinguished:

- *Perfect* secrecy means that an adversary gains absolutely no information about the secret.

- *Statistical* secrecy is also *unconditional*, i.e., unrestricted adversaries are considered, but the adversary is allowed to learn a little about the secret.

- *Computational* secrecy means that the adversary is assumed to be restricted to efficient computations, and currently, computational secrecy always relies on unproven assumptions about the hardness of certain problems.

To make statistical secrecy precise, it is natural to describe B's a priori knowledge about x by a probability distribution p from which he knows that x is chosen. After the protocol, given B's view of the protocol, x has a possibly different distribution q. We can say that B learns almost nothing new if p and q are somehow close to each other. A complicating, but often overlooked factor is that this should be true for any a priori distribution p.

In the following, we consider two formal notions of the closeness of ensembles of probability distributions. One of these, which we call the *bias-based secrecy property*, is based on the L_1-norm difference between p and q. When x consists of just one bit, this describes the additional advantage B obtains in guessing the value of x. Thus it is the natural extension of the existing definition of statistically hiding commitment schemes for one bit. Moreover, it coincides with the definition of statistical zero-knowledge in [17]. The other definition is based on the difference in Shannon entropy between p and q. Thus it describes how much information about x the adversary can learn from the protocol. We call it *capacity-based secrecy property*, because it corresponds to considering the protocol as a channel with x as input and B's view as output. The fact that we can prove this definition to be essentially equivalent to the first one allows for much more elegant proofs of secrecy, for instance, if one considers commitments to many bits or many commitments to the same value.

This last point is illustrated by the proof of secrecy for the commitment scheme we present in this paper. This scheme allows commitment to many bits. Its commitment and reveal protocols have a very small constant number of rounds, and the size of a commitment is independent of the number of bits committed to. This also holds for the total communication complexity, except of course for the bits needed to send the secret when it is revealed.

Most schemes in the literature are just bit commitment schemes, and thus, if one commits to many bits, each bit is expanded to, e.g., 500 bits. Note that efficient multi-bit commitment schemes can also be used to reduce the communication complexity of zero-knowledge protocols [20]. A concrete example of this for Boolean circuit satisfiability is given later in the paper (see Section 5.3).

Unconditionally hiding bit commitment schemes were presented, e.g., in [6, 4, 8]. In these schemes, the fact that A cannot later change the bit committed to relies on specific number-theoretic assumptions. More general assumptions are used in [24], which can be based on any collision-intractable hash function, and [23], based on any one-way permutation. Unconditionally hiding *multi-bit* commitment schemes were presented in [3, 26, 9, 2]. They all rely on specific number-theoretic assumptions, the hardness of computing discrete logarithms or factoring integers. In contrast, our scheme is based on any collision-intractable hash function. This is an improvement in theory, because the assumption is weaker, and useful in practice, because one can use efficient conventional hash functions such as SHA-1 [27] or RIPEMD-160 [14] (follow the references in [14] for more such functions and known attacks). The construction is an improvement of the scheme from [24]. Our construction was presented in preliminary form at Crypto '93. A couple of years later Halevi and Micali, who were apparently unaware of the Crypto '93 result, rediscovered the construction and presented it at Crypto '96.

Naor [22] has also presented a multi-bit commitment scheme with small amortised communication complexity, based on general assumptions. However, that scheme is of a type dual to ours, i.e., it is only computationally hiding, whereas the binding is unconditional. On stronger assumptions, schemes of this type must have been known in the folklore before, e.g., based on efficient probabilistic public-key encryption.

1.1 Organization of the Rest of this Paper

In Section 2, we introduce notation about protocols. In Section 3, we introduce our two main definitions of statistical secrecy. Section 4 shows that these definitions are equivalent except for small transformations of the security parameters. In Section 5, we give a precise definition of multi-bit commitments, present our construction, and prove its security. We show how this and the results of [20, 5] directly give a statistical zero-knowledge argument for Boolean circuit satisfiability with small communication complexity. In Section 6, we present further evidence that our definitions of statistical secrecy are universal: First we show that strengthening the adversaries by auxiliary inputs makes no difference, and sequential composition of statistically secret protocols is therefore possible. Secondly, we consider statistical counterparts of well-known computational secrecy definitions and show that they are also essentially equivalent to our two main definitions.

2 Protocol Notation

The model of protocols used in this paper is based on probabilistic interactive Turing machines as defined in [17]. These are Turing machines equipped with a read-only input tape, a work tape, a random tape, and two communication tapes. One communication tape is write-only and used for sending messages, and the other is read-only and used for receiving messages.

A 2-party protocol is a pair of interactive Turing machines sharing their communication tapes. The *view* of a participant, A, in an execution of an interactive protocol with B is defined to consist of A's input, all random bits used by A, and all the messages sent and received in this execution of the protocol. We refer to [17] for detailed definitions.

All our protocols have a security parameter k. This means that k in unary representation is a common parameter on the input tapes of both participants. Usually, there are secret inputs, too, i.e., parameters that are on only one of the input tapes. The honest participants in our protocols are polynomial-time, i.e., the corresponding interactive Turing machines stop after polynomial time in k.

$\tilde{X}$ will denote any machine playing the role of a machine X in a given protocol, but not necessarily following the prescribed methods. Such machines are used to model cheaters. Note that this does not tacitly restrict "unrestricted" adversaries $\tilde{B}$ to computable functions, because we will quantify over k and $\tilde{B}$ separately.

3 Definitions of Secrecy

We now present definitions of the statistical secrecy of one party's input in a 2-party protocol.

Consider a 2-party protocol, (A, B), with security parameter k. The input of A, apart from the common parameter k, is denoted by x. The question is how much a possibly cheating $\tilde{B}$ learns about x. We assume that x is chosen from a finite set $M(k)$, whose size $N(k) = |M(k)|$ may depend on k. We simply write M and N if no confusion about k is possible. We also assume that $N(k)$ is non-decreasing and that B does not have an input, except for k. This assumption does not reduce the generality of the results of this section, see Section 6.1.

We mostly work in the following probability spaces: Let any $\tilde{B}$, any k, and any a priori distribution p of A's input be given, where p_x denotes the a priori probability of a particular value x. The protocol now induces a joint probability space on the views of both parties, determined by the choice of x and the random choices, i.e., the contents of the random tapes, of both parties. Probabilities in this space are simply denoted by $Pr[\cdot]$. Let X denote the random variable corresponding to x and V the random variable corresponding to the view of $\tilde{B}$. Whenever one of the parameters $\tilde{B}$, k, and p is not clear from the context, X or V will be given corresponding indices, e.g., X_p. Individual views of $\tilde{B}$ are usually denoted by v. Random variables that are clear from the context are omitted, e.g., we write $Pr[x|v]$.

We denote the a posteriori probability of the input x as seen by $\tilde{B}$ after an execution of the protocol with resulting view v by q, where

$$q_x = Pr[x \mid v].$$

For the first definition of statistical secrecy of A's input, we define

$$bias_p(v) = \sum_{x \in M} |p_x - q_x|$$

for each view v. In other words, $bias_p(v)$ is the distance in the L_1-norm, $||p - q||_1$, between the a priori distribution of the secret input and its a posteriori distribution, given the view v. Let

$$Bias_p = \sum_v Pr[v] bias_p(v)$$

be the expected value of $bias_p(v)$.

Definition 3.1 The protocol has the *bias-based secrecy property* if $Bias_p \leq 2^{-k}$ for all distributions, p, all security parameters, k, and for all adversaries, $\tilde{B}$. □

In this definition, $\tilde{B}$ can be specific for a specific p; in other words, $\tilde{B}$ can have arbitrary a priori information about A's input. It is basically a definition about the machine A only.

For the second definition, note that the protocol defines transition probabilities from the secret inputs x to the views v of $\tilde{B}$. If it is considered as a channel with a value of X as input and a value of V as output, its channel capacity $C_{\tilde{B}}$ is

$$C_{\tilde{B}} = \max_p(I(X_p; V_p)),$$

where I denotes the mutual information between two random variables. Recall that mutual information is defined as

$$I(X_p; V_p) = H(X_p) - H(X_p|V_p),$$

where

$$H(X_p) = -\sum_{x \in M} p_x \log_2 p_x$$

is the entropy of X_p and

$$H(X_p|V_p) = -\sum_{x,v} Pr[x,v] \log_2 Pr[x|v]$$

the conditional entropy of X_p given V_p. For these definitions and simple rules for computing with mutual information and entropies, see, e.g., [15, Sections 2.2 and 2.3].

Definition 3.2 The protocol is said to have the *capacity-based secrecy property* if, for every adversary, $\tilde{B}$, and every security parameter, k, its channel capacity $C_{\tilde{B}}$ is at most 2^{-k}. □

This is a natural information-theoretic definition of secrecy, because it means that for any a priori information, $\tilde{B}$'s view from the commit protocol only gives negligible additional information about X.

4 Relations Between the Definitions

We will show next that the bias-based secrecy property and the capacity-based secrecy property are equivalent except for small transformations of the security parameters.

Theorem 4.1 *Consider a two-party protocol, (A, B), as above and let $n(k) = \log_2 N(k)$.*

a) *If (A, B) has the bias-based secrecy property, it has a channel capacity of at most $2^{-k}(n(k) + k)$ for any $\tilde{B}$.*

Thus, a capacity of at most 2^{-k} can be achieved by using a security parameter k' where $k' \geq k + \log_2(n(k') + k')$.

b) *If (A, B) has the capacity-based secrecy property, it has an average bias of at most $\sqrt{2 \ln 2} \cdot 2^{-k/2}$ for any $\tilde{B}$.*

Thus, a bias of at most 2^{-k} can be achieved by using a security parameter $k' \geq 2k + 1$.

The two parts of this theorem are proved in the following two subsections.

We also show that the bounds given in both parts are essentially optimal, see the examples at the end of both subsections. In particular, Example 4.1shows that there is no bound on the capacity given just the bias, independent of $n(k)$. This may be the reason why no related inequalities seem to exist in the literature, in contrast to Part b). Nevertheless, the condition on k' in Part a) of the theorem is always solvable, unless $n(k')$, the number of bits of the secret, is growing exponentially in k', which would be unreasonable in practice.

4.1 From Bias to Capacity

We start with some preliminaries about the entropy function.

Definition 4.2 Define $0 \log_2 0 = 0$ according to usual conventions. For any finite set M, we define a continuation of the entropy function H on all functions $p : M \to \mathbb{R}^+ \cup \{0\}$ (i.e., not only those that are probability distributions) by the standard equation

$$H(p) = -\sum_{x \in M} p_x \log_2 p_x,$$

where p_x is short for $p(x)$. □

Note that we have now defined entropy directly on probability distributions, not only via random variables that induce these distributions. In the following we shall freely use H to denote the entropy in both cases. Thus $H(p) = H(X_p)$ in the scenario of Section 3.

Lemma 4.3 *Let us define a function* $H_1(x) = -x \log_2(x)$ *on* $\mathrm{I\!R}^+ \cup \{0\}$. *Then:*

- H_1 *only has zeros at* $x = 0$ *and* $x = 1$.
- H_1 *is convex-*$\cap$ *and has a maximum at* e^{-1}.
- $H_1(x) \geq H_1(1-x)$ *for* $0 \leq x < 1/2$.

Proof The first part is obvious. For the second part, note that the derivative is $H_1'(x) = -log_2(x) - c$ with $c = \log_2(e)$. Thus $H_1'(x)$ is monotonic decreasing with a zero at $x = e^{-1}$.

The third part is clear for $e^{-1} \leq x < 1/2$, because $H_1(x)$ is monotonic decreasing in this interval. Thus we now consider $x < e^{-1}$. As $H_1(x)$ is convex-$\cap$, it lies above the line through (0, 0) and $(e^{-1}, H_1(e^{-1}))$ everywhere in this interval. The gradient of this line is $-e^{-1}\log_2(e^{-1})/e^{-1} = c$. Similarly, $H_1(1-x)$ lies below the tangent to H_1 at (1, 0) for all $x > 0$. The gradient of this tangent is $H_1'(1) = -c$. Thus $H_1(x) > cx > H_1(1-x)$ for $0 < x < e^{-1}$. Finally, the case $x = 0$ is obvious.

□

Lemma 4.4 *Let* f *be a differentiable convex-*$\cap$ *function on an interval* $[a, b] \subset \mathrm{I\!R}$, *and let a distance* $\epsilon > 0$ *be given. Then the difference* $|f(x) - f(x')|$ *for values* $x, x' \in [a, b]$ *with* $x - x' = \epsilon$ *is maximal at one of the two boundaries, i.e., the maximum is either* $|f(a+\epsilon) - f(a)|$ *or* $|f(b) - f(b-\epsilon)|$.

Proof We consider the function $g(x) = f(x+\epsilon) - f(x)$ on $[a, b-\epsilon]$. Its derivative, $g'(x) = f'(x+\epsilon) - f'(x)$, is negative, because f' is monotonic decreasing. Thus g is monotonic decreasing, and $|g|$ is maximal at one of the boundaries. □

Lemma 4.5 *For any probability distributions* p, q *on a finite set* M,

$$|H(p) - H(q)| \leq H(|p - q|),$$

where the absolute value of a distribution is taken component-wise.

Proof We have

$$|H(p) - H(q)| = |\sum_{x \in M} (p_x \log_2(p_x) - q_x \log_2(q_x))|.$$

We show the desired inequality pointwise, i.e., with $\epsilon_x = |p_x - q_x|$, we show

$$|p_x \log_2(p_x) - q_x \log_2(q_x)| \leq -\epsilon_x \log_2(\epsilon_x) \qquad (*)$$

for all x. For symmetry reasons, it suffices to consider $p_x > q_x$. Note that we are considering differences between two values $H_1(p_x)$ and $H_1(q_x)$. As H_1 is convex-$\cap$ by Lemma 4.3, the differences are maximal at either of the boundaries by Lemma 4.4. These two possibilities are

- $|H_1(\epsilon) - H_1(0)| = H_1(\epsilon)$ and
- $|H_1(1) - H_1(1 - \epsilon)| = H_1(1 - \epsilon)$.

By Lemma 4.3, the first one is larger. This proves $(*)$ and thus the lemma.

□

In the final lemma, we bound "entropies" $H(\epsilon)$ of functions that have small L_1-norm:

Lemma 4.6 *If* $||\epsilon||_1 = d$ *for a function* $\epsilon : M \rightarrow \mathrm{I\!R}^+ \cup \{0\}$*, then*

$$H(\epsilon) \leq d(n - \log_2(d)),$$

where $n = \log_2 |M|$.

Proof We scale ϵ to L_1-norm 1, i.e., to a probability distribution, and exploit that the entropy of probability distributions on a given set is maximal for the uniform distribution. Let $E = d^{-1}\epsilon$. Then

$$\begin{array}{rcl} H(\epsilon) & = & H(dE) \\ & = & -\sum_{x \in M} d \cdot E_x \log_2(d \cdot E_x) \\ & = & -d(\sum_{x \in M} E_x(\log_2(E_x) + \log_2(d))) \\ & = & d(H(E) - \log_2(d) \sum_{x \in M} E_x) \\ & = & d(H(E) - \log_2(d)) \\ & \leq & d(n - \log_2(d)). \end{array}$$

This proves the lemma. □

Proof of Theorem 4.1 a) We fix $\tilde{B}$ and p and prove $I(X_p;V_p) \leq 2^{-k}(n(k)+k)$ from the precondition $Bias_p \leq 2^{-k}$.

In this proof, we denote the distribution of X_p, given that $V = v$, by $q^{(v)}$, to distinguish different v's. We have

$$\begin{aligned} I(X_p;V_p) &= \sum_v Pr[v](H(X_p) - H(X_p|v)) \\ &\leq \sum_v Pr[v]|H(p) - H(q^{(v)})| \\ &\leq \sum_v Pr[v]H(|p - q^{(v)}|) \qquad \text{by Lemma 4.5.} \end{aligned}$$

We now apply Lemma 4.6. Note that $d = ||p - q^{(v)}||_1$ is just $bias_p(v)$. Let f be defined by $f(d) = d(n(k) - \log_2(d))$. Then we have

$$I(X_p;V_p) \leq \sum_v Pr[v]f(bias_p(v)).$$

The function f is easily seen to be convex-$\cap$ for $d \geq 0$. Thus we can apply Jensen's inequality. Note that the average over v in the formula above equals the average over $d = bias_p(v)$.

$$\begin{aligned} I(X_p;V_p) &\leq f(\sum_v Pr[v]bias_p(v)) \\ &= f(Bias_p) \\ &\leq 2^{-k}(n(k)+k). \end{aligned}$$

This is the desired result. □

The following example shows that it is not in general possible to remove the dependency on $n(k)$ from the bound we have proved.

Example 4.1

Consider the protocol in which the secret input is simply revealed. Then, it is easy to see that both the average bias and the information are maximal when the input is uniformly distributed. In this case, the average bias is $2 - 2/N(k)$ and the information is $\log_2 N(k) = n(k)$. Thus the capacity is approximately $n(k)/2$ times the maximal bias. The above bound is therefore optimal except for a factor of 2 and the addition of k. Furthermore, this protocol also shows that the inequality

$$I(X_p; V_p) \leq Bias_p \cdot (n(k) + k),$$

which is obtained in the above proof, is optimal within a factor of 2. $\Box$

4.2 From Capacity to Bias

We will prove the second half of Theorem 4.1 using Pinsker's inequality. Let μ, η be probability distributions on a finite set with probabilities μ_i, η_i. Then the *information divergence* from μ to η is defined as

$$D(\mu||\eta) = \sum_i \mu_i \log_2 \frac{\mu_i}{\eta_i},$$

where $\mu_i \log_2(\mu_i/\eta_i)$ is defined to be 0 if $\mu_i = 0$ and ∞ if $\mu_i > \eta_i = 0$.

Theorem 4.7 *Pinsker's Inequality: The information divergence is related to the L_1-norm distance between the two probability distributions as follows:*

$$D(\mu||\eta) \geq \frac{1}{2\ln 2}||\mu - \eta||_1^2.$$

For background on these results, see [11, p.20 and p.58-59]. For our protocol situation, we can derive the following lemma:

Lemma 4.8 *For any distribution p, we have that*

$$I(X_p; V_p) \geq \frac{1}{2\ln 2} Bias_p^2.$$

Proof Let μ be the joint distribution of X_p and V_p, and let η be the distribution we would have if X_p and V_p were independent, i.e., $\eta_{x,v} = Pr[X_p = x]Pr[V_p = v]$. On the one hand, it can easily be seen that $D(\mu||\eta) = I(X_p; V_p)$ from the respective definitions. On the other hand,

$$\begin{aligned} ||\mu - \eta||_1 &= \sum_{x,v} |Pr[X_p = x, V_p = v] - Pr[X_p = x]Pr[V_p = v]| \\ &= \sum_v Pr[V_p = v] \sum_x |Pr[X_p = x|V_p = v] - Pr[X_p = x]| \\ &= Bias_p. \end{aligned}$$

Thus, the lemma follows from Pinsker's inequality. □

Theorem 4.1 b) is an immediate consequence of this lemma.

The following example shows that it is really necessary to double the security parameter when going from capacity to bias.

Example 4.2
We consider a protocol that implements a binary symmetric channel with an error probability of $\frac{1}{2} - \epsilon$, independent of B and thus of $\tilde{B}$. Hence we have two inputs x_0, x_1 and two views v_0, v_1 such that $Pr[v_i|x_i] = 1/2 + \epsilon$. The capacity of this channel is well-known to be

$$\begin{aligned}
C &= 1 - H(\frac{1}{2} - \epsilon) \\
&= 1 + (\frac{1}{2} - \epsilon)\log_2(\frac{1}{2} - \epsilon) + (\frac{1}{2} + \epsilon)\log_2(\frac{1}{2} + \epsilon) \\
&= 1 + (\frac{1}{2} - \epsilon)(\log_2(1 - 2\epsilon) - 1) + (\frac{1}{2} + \epsilon)(\log_2(1 + 2\epsilon) - 1) \\
&= (\frac{1}{2} - \epsilon)\log_2(1 - 2\epsilon) + (\frac{1}{2} + \epsilon)\log_2(1 + 2\epsilon) \\
&= \frac{1}{ln(2)}((\frac{1}{2} + \epsilon)\ln(1 + 2\epsilon) + (\frac{1}{2} - \epsilon)\ln(1 - 2\epsilon)) \\
&\leq \frac{1}{ln(2)}((\frac{1}{2} + \epsilon)2\epsilon + (\frac{1}{2} - \epsilon)(-2\epsilon)) \\
&= \frac{1}{ln(2)}4\epsilon^2.
\end{aligned}$$

We now compute $Bias_p$ for the case where p is the uniform distribution on x_0, x_1. For any x and v, the a posteriori probability q_x equals $\frac{1}{2} + \epsilon$ or $\frac{1}{2} - \epsilon$. Thus

$$Bias_p = \sum_v Pr[v] \sum_x |p_x - q_x| = 2 \cdot \frac{1}{2} \cdot 2 \cdot \epsilon = 2\epsilon.$$

This implies

$$C \leq \frac{1}{ln(2)} Bias_p^2,$$

which means that the bound in Theorem 4.1 b) is tight except possibly for a factor of 2. □

5 Commitment to Many Bits

5.1 Definitions of Multi-Bit Commitments

We define a multi-bit commitment scheme as a triple (*commit*, *reveal*, n), where *commit* and *reveal* are two-party protocols and $n : \mathbb{N} \to \mathbb{N}$ is a function denoting the length of the strings that can be committed to. The protocols take place between parties A and B, where A is the party committing herself.

- The commit protocol, (A_c, B_c), has a security parameter k, and A_c gets a secret value x from $\{0,1\}^{n(k)}$ as input. The concatenation of all messages sent in an execution of the commit protocol is called *the commitment*.

  Either party may reject in the commit protocol, but if both parties are honest, this should almost never happen. More formally, A_c and B_c may output a special value *reject*, but if in fact (A_c, B_c) is executed, and not $\tilde{A}_c$ or $\tilde{B}_c$, the probability that an output is *reject* decreases faster than k^{-c} for all $c > 0$.

- The reveal protocol is denoted (A_r, B_r). The input of A_r should be the view of A_c in the commit protocol, while B_r gets the commitment as input. At the end of the reveal protocol, B_r outputs *reject* or a pair (*accept*, x). The intuitive meaning is that either B has detected cheating by A, or he accepts that A has opened the commitment to reveal the value x.

In some concrete schemes, it makes sense to define the commitment and the input to A_r as a subset or a function of the messages sent and the view, respectively. However, our definition is simpler and without loss of generality.

We will only consider commitment schemes with *non-interactive opening*, i.e., where the reveal protocol consists of A_r sending one message to B_r. Without loss of generality, we can then assume that the message is of the form (x, m), where x should be the value committed to, and that B_r, on receiving such a message from A_r or $\tilde{A}_r$, never outputs (*accept, x'*) with $x' \neq x$.

We have already built into the model another useful property that our construction fulfils and that we call *public verification*: B can verify the opening based on the commitment only. This means that anyone who trusts that a given commitment is the result of a conversation with B can verify the opening without knowing B's random bits.

Note that the equivalence results in the preceding section also hold for commitment schemes without these two properties.

Definition 5.1 A pair of protocols as described above is called a multi-bit commitment scheme with non-interactive opening if it has the following two properties:

- *Binding property*: Let $\tilde{A}_c$ be any polynomial-time bounded machine that executes the commit protocol with B and then outputs two messages (x, m) and (x', m'). Intuitively, with these messages, the cheating committer hopes to have the choice between opening the commitment to reveal x or x', respectively. Let $p(\tilde{A}_c, k)$ be the success probability of $\tilde{A}_c$, i.e., the probability that both messages (x, m) and (x', m') would be accepted by B_r. The probability is taken over the coin-flips of $\tilde{A}_c$, B_c and the two executions of B_r. Then $p(\tilde{A}_c, k) < k^{-c}$ for all $c > 0$ and k sufficiently large.

- *Statistically hiding*: A multi-bit commitment scheme is called *statistically hiding* if its protocol *commit* has

  - the bias-based secrecy property, see Definition 3.1, or
  - the capacity-based secrecy property, see Definition 3.2.

□

By Theorem 4.1, the two possibilities in the secrecy definition are equivalent except for small transformations of the security parameter. Recall that these definitions assume that $\tilde{B}$ can have arbitrary a priori information about the string committed to.

5.2 Efficient Statistically Hiding Commitments

Naor and Yung have shown that a statistically hiding bit commitment scheme can be built from collision-intractable hash functions [24]. This scheme needs interaction only in an initialisation phase, after which both committing and opening are non-interactive. We now modify this scheme to get efficient multi-bit commitments. The amortised number of bits of communication per bit committed to is only O(1). Our scheme makes use of families of universal$_2$ hash functions, defined in [7]:

Definition 5.2 A class F of functions $A \rightarrow B$, where A and B are finite sets, is called universal$_2$ if for any distinct $a_1, a_2 \in A$ the probability that $f(a_1) = f(a_2)$ is at most $1/|B|$, when f is chosen uniformly at random in F. □

In practice we need a family $F = \{F_m\}_{m \in \mathbb{N}}$ of universal$_2$ classes of functions $\{0,1\}^m \rightarrow B_m$ such that the random choice of a function $f \in F_m$, given m, and the evaluation of f can be done in polynomial time in m.

In particular, our construction will be efficient if we use the following functions, essentially from [7].

Lemma 5.3 *Let m-bit strings be identified with elements of $GF(2^m)$. Each of the following classes F_m of functions from m-bit strings to i-bit strings (with $i \leq m$) is universal$_2$:*

$$F_m = \{f : \{0,1\}^m \rightarrow \{0,1\}^i; z \mapsto az + b|_i \mid a, b \in GF(2^m)\}.$$

Here $|_i$ means taking the i least significant bits.

Moreover, we need collision-intractable hash functions, defined in [12].

Definition 5.4 A family of collision-intractable hash functions is a family H of finite sets $\{H_m\}_{m \in \mathbb{N}}$ with the following properties: Each $h \in H_m$ is a function $\{0,1\}^m \rightarrow \{0,1\}^{l(m)}$, where $l(m) : \mathbb{N} \rightarrow \mathbb{N}$ is a function with $l(m) < m$ for all m. Both the random choice of a function $h \in H_m$, given m, and the evaluation of h are possible in polynomial time in m.

Finally, collision-intractability means that for all $c > 0$ and all probabilistic polynomial-time algorithms A_H, the probability that A_H finds $x, y \in \{0,1\}^m$ such that $x \neq y$ and $h(x) = h(y)$ is less than m^{-c} for m sufficiently large. The probability is over the random selection of h and the random choices of A_H. □

In the commitment scheme, we need a family H^* such that the output length of functions in H_k^* for a security parameter k is $k + 1$ and the input length is arbitrary, as long as polynomial in k. By [13], such a collision-intractable family of functions can be constructed from the functions with the input and output lengths as in Definition 5.4.

Now we propose the following commitment scheme:

Protocol 1

Initialisation Phase

Let $k' = k+1$. B_c chooses a random hash function $h \in H_k^*$, i.e., $h : \{0,1\}^+ \to \{0,1\}^{k'}$, and sends it to A_c.

Commit Protocol

On input an n-bit string x, A_c chooses at random a $3k'$-bit string y and a universal$_2$ hash function f from $3k'$ bits to k' bits. It sends

$$c = h(f||h(y)||h(x) \oplus f(y))$$

to B_c, where $\oplus$ denotes the bitwise XOR and $||$ denotes the concatenation of bit strings. Intuitively, $h(x) \oplus f(y)$ is $h(x)$ encrypted with a so-called "privacy amplified" version of y.

Reveal Protocol

A_r sends (f, x, y) to B_r. B_r checks that $c = h(f||h(y)||h(x) \oplus f(y))$. If yes, it outputs (*accept*, x), otherwise *reject*. □

For the analysis of this protocol, we use the following privacy amplification result (see Fig. 1).

Theorem 5.5 *Let $y \in \{0,1\}^{k''}$ be chosen uniformly at random, and let $e : \{0,1\}^{k''} \to \{0,1\}^t$ be an arbitrary function. Let $0 < s < k''-t, k' = k''-t-s$ and let F be a universal$_2$ class of hash functions from $\{0,1\}^{k''}$ to $\{0,1\}^{k'}$. If f is chosen uniformly at random in F, the expected entropy of $f(y)$ when f, e and $e(y)$ are given is at least $k' - 2^{-s}/\ln 2$ bits. More formally,*

$$H(F(Y)|F, e(Y)) \geq k' - \frac{2^{-s}}{\ln 2}.$$

Here F denotes the random variable defined by the choice of f (no confusion with the set F should be possible), and e is fixed and thus not listed in the condition.

This theorem is almost a restatement of [1, Corollary 5] which in our notation is $I(F(Y); F, e(Y)) \leq 2^{-s}/\ln 2$. One can easily see in the proof of that corollary that our slightly stronger statement is also proved.

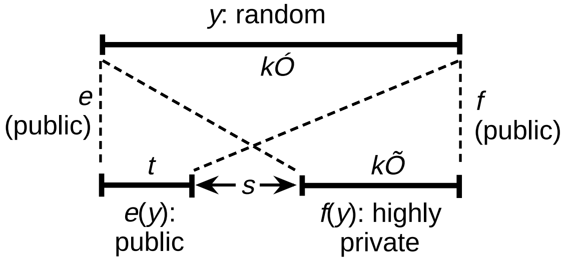

Figure 1: Privacy amplification: y is completely random, but some information is given by $e(y)$. s is a security margin.

Theorem 5.6 *Protocol 1 is a statistically hiding commitment scheme, under the assumption that the family H is collision-intractable. It allows committing to n bits by a commitment of size $k+1$ bits and total communication complexity for commitment and opening of $10(k+1)$ bits, plus the n bits of x.*

One or more parties A can execute the protocol commit *an arbitrary, more precisely polynomial in k, number of times with B based on one execution of the initialisation phase.*

Proof The size of the commitments and total communication complexity is clear from the description above and the fact that the universal$_2$ hash function can be specified by $6k'$ bits. We did not count the initialisation phase here, because we assume that it has been carried out once for many commitments. Anyway, with most proposed collision-intractable families of hash functions, a function h can also be specified with $d \cdot k$ bits, where d is a small integer constant.

The binding property is an immediate consequence of the collision intractability of H: Assume that an algorithm $\tilde{A}_c$ contradicts the binding property. Then construct a collision-searching algorithm A_H as follows: On input h, it simply calls $\tilde{A}_c$, which is also a non-interactive algorithm that works on an input h from the initialisation phase. Suppose $\tilde{A}_c$ outputs c and $(f, x, y) \neq (f', x', y')$ which both open the commitment correctly, and where $x \neq x'$ (this is necessary for $\tilde{A}_c$'s success). It follows that $h(f'||h(y')||h(x') \oplus f'(y')) = c = h(f||h(y)||h(x) \oplus f(y))$, so if $(f'||h(y')||h(x') \oplus f'(y')) \neq (f||h(y)||h(x) \oplus f(y))$,

A_H outputs these two values. Otherwise it follows that $h(y) = h(y')$. Thus, A_H outputs (y, y') if $y \neq y'$. Otherwise, we know $y = y'$ and $f = f'$, and thus $h(x) = h(x')$. Therefore A_H now outputs (x, x'), where $x \neq x'$ by the initial assumption. It follows from this description that the success probability of A_H is exactly the same as that of $\tilde{A}_c$.

We now show capacity-based secrecy. First observe that it is enough to show this for a modified commit protocol in which $f, h(y)$ and $h(x) \oplus f(y)$ are sent, since this gives the recipient even more information than before. Then let an a priori distribution of x and an arbitrary $\tilde{B}_c$ be given. Applying Theorem 5.5 with $e = h$, $k'' = 3k'$, and $t = s = k'$, and thus in fact $k' = k'' - t - s$, gives a bound on $\tilde{B}_c$'s expected uncertainty about $f(y)$ if it knows f, h, and $h(y)$:

$$H(F(Y)|F, h(Y)) \geq k' - \frac{2^{-k'}}{\ln 2} > k' - 2^{-k}.$$

The view v of $\tilde{B}_c$ in the modified commit protocol consists of just these values $v' = (f, h, h(y))$ and $h(x) \oplus f(y)$. Thus we see that $\tilde{B}_c$ also has very little information about X: Let $Z = F(Y)$; then

$$\begin{aligned}
I(V; X) &= I(V', h(X) \oplus Z; X) \\
&= I(V'; X) + I(h(X) \oplus Z; X|V') \\
&= 0 + H(h(X) \oplus Z|V') - H(h(X) \oplus Z|X, V') \text{ (indep. of } V' \text{ and } X) \\
&\leq k' - H(h(X) \oplus Z, X, Z|X, V') \qquad (1) \\
&= k' - H(Z|X, V') \qquad (2) \\
&= k' - H(Z|V') \qquad \text{(indep. of } X \text{ and } V', Z) \\
&\leq k' - (k' - 2^{-k}) = 2^{-k}.
\end{aligned}$$

In (1) we used that X and Z are functions of $h(X) \oplus Z$ and X; similarly in (2).

As this bound holds for all a priori distributions, it is also a bound on the capacity.

Note that both parts of this proof are still valid if A makes more than one commitment based on the same public function h. □

One can easily see that among the three applications of h in the commit protocol, only that to y is essential for security. Hashing x may be omitted if x is rather short anyway. The final hashing of the commitment may be omitted in applications where the efficiency of the reveal protocol seems more important than that of the commit protocol. The given version with very short commitments and longer revealing is particularly suitable if not all commitments are opened.

5.3 Using the Commitment Scheme to Build Zero-Knowledge Protocols

This subsection considers an application of our commitment scheme to construct zero-knowledge protocols. We assume here that the reader is familiar with the concepts of proof systems and zero-knowledge. For formal definitions please refer to [17]. An *interactive argument* is the same as a proof system, except that the soundness property is only required to hold for all polynomial time cheatingprovers.

Our commitment scheme allows us to build a statistical zero-knowledge argument for Boolean circuit satisfiability, and so for any NP problem. This can be seen by combining the scheme with two other ingredients:

- The protocol by Brassard et al. from [5] for showing that a Boolean circuit is satisfiable. This protocol works based on any bit commitment scheme for single bits and is a computational zero-knowledge proof system or a perfect/statistical zero-knowledge argument, depending on whether the commitments used are computationally or unconditionally hiding. The basic step in the protocol is that the prover commits to $O(n)$ bits, where n is the size of the circuit, and depending on a random challenge from the verifier, the prover either opens all the bits or a specific subset of them that depends on the satisfying assignment. This basic step is iterated a number of times.

- The method by Kilian et al. from [20] for using a multi-bit commitment scheme in any protocol of a type they call “subset-revealing”, of which the protocol from [5] is an example. The interesting point is that the method works even though the commitment scheme does not allow opening individual bits in a multi-bit commitment. The method replaces each basic step in the original protocol by a new one which needs 5 messages instead of 3 and contains 2 commitments to $O(n)$ bits each instead of $O(n)$ commitments to 1 bit each. If the prover could cheat in the old basic step with probability 1/2, he can cheat in the new one with probability 3/4.

By combining these three ingredients, one obtains an extremely efficient statistical zero-knowledge argument for Boolean circuit satisfiability, and hence for any NP problem. More precisely, one can prove the following theorem.

Theorem 5.7 *Assume that a family of collision-intractable hash functions exists. Then there is a statistical zero-knowledge argument for Boolean circuit satisfiability with the following properties: if the input circuit is of size n, then the protocol requires communicating $O(n^2)$ bits. If any probabilistic polynomial-time prover can cheat with probability $\epsilon(n) \geq 2^{-n}$, then there is a probabilistic algorithm that can find collisions for the hash function used in expected time polynomial in n and proportional to $1/\epsilon(n)^2$.*

Note that for a protocol of the type we consider, there are actually a number of parameters, which one may consider *independently*: the size of the input circuit, n, the logarithm of the probability with which we will allow the prover to cheat (assuming he cannot break the hash function), and the output length of the hash function. To simplify, we have followed a number of earlier works in the theorem above and have let all parameters be $O(n)$.

Using the protocol from [5] based on a 1-bit commitment scheme would give a communication complexity of $O(n^3)$ bits. Kilian [18, 19] has found a protocol based on probabilistically checkable proofs that would, with our choice of parameters, have a communication complexity of $O(n^2 \log n)$ [1]. Using a completely different method, Cramer and Damgård [10] obtained an argument that also has $O(n^2)$ complexity. In comparison, their protocol is perfect zero-knowledge and constant round, but it is based on more specialized assumptions, namely the hardness of computing discrete logarithms in a group of prime order or of factoring integers.

Perhaps even more interesting is the performance in practice. For instance, if we use SHA-1 as the hash function, which has a 160-bit output, and we set the maximal probability for the prover to cheat at 2^{-50}, then a

[1] But our protocol would not be superior to Kilian's for all choices of parameters – in fact Kilian shows that the communication complexity does not have to depend on n at all.

circuit consisting of 10000 gates could be proved satisfiable using about 3 Mbyte of communication.

To assess the computation effort required, it seems reasonable to assume that an implementation would spend almost all its time hashing. SHA-1 can be implemented on standard PC's at speeds around 6-8 Mbyte/sec. This suggests that, at a security level of 2^{-50}, a real implementation should be able to handle around 20000 gates per second, assuming that the communication lines can keep up. To the best of our knowledge this is the most practical protocol proposed for circuit satisfiability.

6 More Variants of Statistical Secrecy

6.1 Auxiliary Input and Composition

In Section 3, we have defined secrecy of an input x against an adversary $\tilde{B}$ that has no input, or at least none related to x. Such inputs would be called *auxiliary inputs.* In computational zero-knowledge, including auxiliary inputs in the definition proved necessary for the secrecy if a protocol is executed repeatedly [25, 28]. Similarly, such auxiliary inputs occur if a statistically hiding protocol is repeated. However, in this case we can show quite easily that secrecy in the setting with auxiliary input is a consequence of normal secrecy. We now describe this formally.

An auxiliary-input attacker $\tilde{B}_{aux}$ on a two-party protocol (A, B) is defined just like a normal attacking $\tilde{B}$, except that $\tilde{B}_{aux}$ also gets an input y, where x and y may have an arbitrary joint a priori distribution p_{aux}. The intuitive idea is that y may be an output from a previous protocol that A executed with the secret x. We define *auxiliary-input capacity* of such a protocol as follows: Let V_{aux} denote the view of $\tilde{B}_{aux}$.

$$C_{\tilde{B}_{aux}} = \max_{p_{aux}} I(V_{p_{aux}}; X_{p_{aux}} | Y_{p_{aux}}),$$

where indices $\tilde{B}_{aux}$ have been omitted for brevity. *Auxiliary-input secrecy* is defined to mean that $C_{\tilde{B}_{aux}} \leq 2^{-k}$ for all $\tilde{B}_{aux}$ and all k.

Lemma 6.1 *If a protocol (A, B) has the capacity-based secrecy property, it also provides auxiliary-input secrecy.*

Proof Let $k, \tilde{B}_{aux}$, and p_{aux} be fixed. We have to show that

$$I(V_{p_{aux}}; X_{p_{aux}} | Y_{p_{aux}}) \leq 2^{-k}.$$

For each y, we define $\tilde{B}_y = \tilde{B}_{aux}(y)$. Thus $\tilde{B}_y$ is a cheater without auxiliary input that acts like $\tilde{B}_{aux}$ on input y. Furthermore, let p_y denote the conditional distribution of x given y. Now,

$$\begin{aligned} I(V_{p_{aux}}; X_{p_{aux}} | Y_{p_{aux}}) &= \sum_y p_{aux}(y) I(V_{p_{aux}}; X_{p_{aux}} | y) \\ &= \sum_y p_{aux}(y) I(V_{\tilde{B}_y, p_y}; X_{p_y}) \\ &\leq \sum_y p_{aux}(y) 2^{-k} \\ &= 2^{-k}. \end{aligned}$$

In the second line, we used that under the condition $Y = y$, the a priori distribution of x is p_y and $\tilde{B}_{aux}$ acts precisely like $\tilde{B}_y$. □

Now we derive a lemma on sequential composability.

Lemma 6.2 *Assume that several 2-party protocols $(A_1, B_1) \ldots, (A_m, B_m)$ are executed sequentially, where any number of them may be equal. As the most general case, we assume that the input of each A_i is some function $f_i(x)$ of one secret x. If all protocols have the capacity-based secrecy property, the joint protocol $(A^*, B^*) = ((A_1, \ldots, A_m), (B_1, \ldots, B_m))$ has a capacity of at most $m \cdot 2^{-k}$.*

Proof Let a probability distribution p of x and an attacker $\tilde{B}^*$ on the joint protocol be given. Without loss of generality, we can split $\tilde{B}^*$ into separate attackers $(\tilde{B}_1, \ldots, \tilde{B}_m)$ on the individual stages $A_1, \ldots, A_m$, where each $\tilde{B}_i$ starts with the views $v_1, \ldots, v_{i-1}$ of the previous stages as an input. Now

$$\begin{aligned} I(V_{\tilde{B}^*,p}; X_p) &= I(V_{\tilde{B}_1}, V_{\tilde{B}_2}, \ldots, V_{\tilde{B}_m}; X_p) \\ &= I(V_{\tilde{B}_1}; X_p) + I(V_{\tilde{B}_2}; X_p | V_{\tilde{B}_1}) + .. + I(V_{\tilde{B}_m}; X_p | V_{\tilde{B}_1}, .., V_{\tilde{B}_{m-1}}) \\ &\leq m \cdot 2^{-k}, \end{aligned}$$

because the i-th summand is bounded by the auxiliary-input capacity of the protocol (A_i, B_i). □

6.2 Counterparts of Computational Secrecy Definitions

For completeness, we now consider statistical counterparts of the most common definitions of computational secrecy and show that they are equivalent to the definitions in Section 3 except for small parameter transformations. For an overview of such definitions for encryption schemes and computational relations between them, see [21].

One of these computational definitions is the so-called polynomial security from [16]. The adversary has to distinguish between only two possible secret inputs x_0 and x_1, among which A chooses the actual secret with probability 1/2 each. The adversary may even choose x_0 and x_1 himself, i.e., pick those that seem easiest to distinguish. Nevertheless, he should not have a significant advantage over mere guessing. In a statistical setting, the fact that the two possible secrets may be specific for each adversary is simply expressed by quantifying over (x_0, x_1) and $\tilde{B}$ separately.

The choice of x_0, x_1 corresponds to the choice of a specific probability distribution p, the uniform distribution on $\{x_0, x_1\}$. The adversary's best strategy is to deterministically make the maximum-likelihood guess given his view v, i.e., guess x_i with $q_{x_i} \geq 1/2$. The probability that this guess is correct is q_{x_i}, and thus we define his advantage over mere guessing as

$$adv(x_0, x_1, v) = |q_{x_0} - \frac{1}{2}| = |q_{x_1} - \frac{1}{2}|.$$

Let

$$Adv(x_0, x_1) = \sum_v Pr[v] adv(v)$$

be the expected value of $adv(x_0, x_1, v)$.

Definition 6.3 The protocol is said to have the *advantage-based secrecy property* if for every adversary, $\tilde{B}$, every security parameter, k, and every pair (x_0, x_1), the average advantage, $Adv(x_0, x_1)$, is at most 2^{-k}. □

Theorem 6.4 *Consider a two-party protocol, (A, B), as above.*

a) *If (A, B) has the bias-based secrecy property, it has an average advantage of at most 2^{-k-1} for all pairs (x_0, x_1).*

b) *If (A, B) has the advantage-based secrecy property, it has a bias of at most 2^{-k+2} for all distributions p.*

Proof a) We fix $\tilde{B}$, x_0, x_1, and k. Let p be the uniform distribution on $\{x_0, x_1\}$. For any view v,

$$\begin{aligned} bias_p(v) &= |p_{x_0} - q_{x_0}| + |p_{x_1} - q_{x_1}| \\ &= 2|\frac{1}{2} - q_{x_0}| \\ &= 2 \cdot adv(x_0, x_1, v). \end{aligned}$$

Thus $Adv(x_0, x_1) = \frac{1}{2} Bias_p$.

b) By the proof of Part a), we know that the bias is at most 2^{-k+1} for all uniform distributions on two values. Now we have to show that it is at most 2^{-k+2} for all distributions p. For any $\tilde{B}$, p, and k, we can rewrite the bias as

$$\begin{aligned} Bias_p &= \sum_v Pr[v] \sum_x |Pr[x] - Pr[x|v]| \\ &= \sum_v \sum_x |Pr[v]Pr[x] - Pr[x, v]| \\ &= \sum_x Pr[x] \sum_v |Pr[v] - Pr[v|x]| \\ &= \sum_x Pr[x] \sum_v |\sum_{x'} Pr[x'](Pr[v|x'] - Pr[v|x])|. \end{aligned}$$

These probabilities are in the probability space induced by p. However, the transition probability $Pr[v|x]$ for a secret input x and a view v only depends on the protocol, i.e., on $\tilde{B}$ and k, and not on p. Hence we can bound the inner sum for any given pair (x, x') by using the precondition about the bias for the uniform definition p' on these two values in the same rewritten form:

$$\begin{aligned} 2^{-k+1} &\geq Bias_{p'} \\ &= 2 \cdot \frac{1}{2} \sum_v |\frac{1}{2}(Pr[v|x'] - Pr[v|x])|. \end{aligned}$$

Substituting this into the expression for $Bias_p$ gives

$$\begin{aligned} Bias_p &\leq \sum_x \sum_{x'} Pr[x]Pr[x'] \sum_v |Pr[v|x'] - Pr[v|x]| \\ &\leq \sum_x \sum_{x'} Pr[x]Pr[x'] 2^{-k+2} \\ &= 2^{-k+2}. \end{aligned}$$

This finishes the proof. $\square$

The second definition of computational secrecy is semantic security, also defined in [16]. Here, an arbitrary efficiently computable function f on the message space is given, and the adversary tries to guess $f(x)$ significantly better given his view v than he could have done a priori. For a statistical counterpart, we simply consider all functions f on the message space. If p and k are fixed, the best a priori guess is a value y such that the probability $Pr[f^{-1}(y)]$ of the preimage set of y is maximal. Similarly, the best a posteriori guess given the view v of the adversary $\tilde{B}$ is a value y such that $Pr[f^{-1}(y)|v]$ is maximal. Thus we define the semantic advantage for each view v as

$$sem_p(f,v) = \max_y Pr[f^{-1}(y)|v] - \max_y Pr[f^{-1}(y)]$$

and its average as

$$Sem_p(f) = \sum_v Pr[v] sem_p(f,v).$$

The corresponding definition of statistical semantic secrecy is clear. It is also clear that advantage-based secrecy is a special case of statistical semantic secrecy where p is the uniform distribution on two messages and f is the identity function. Conversely, we prove that bias-based secrecy implies statistical semantic secrecy. We fix $\tilde{B}$, p, k, and a view v. Let y_{max} be the best a posteriori guess. Then

$$\begin{aligned} sem_p(f,v) &= Pr[f^{-1}(y_{max})|v] - \max_y Pr[f^{-1}(y)] \\ &\leq Pr[f^{-1}(y_{max})|v] - Pr[f^{-1}(y_{max})] \\ &= \sum_{x \in f^{-1}(y_{max})} q_x - \sum_{x \in f^{-1}(y_{max})} p_x \\ &\leq \sum_{x \in f^{-1}(y_{max})} |q_x - p_x| \\ &\leq \sum_{x \in M} |q_x - p_x| \\ &= bias_p(v). \end{aligned}$$

Averaging over v gives the desired result.

The third well-known definition of computational secrecy is Yao's in [29]. It can be seen as a computational counterpart of capacity-based secrecy.

7 Conclusion

We have studied notions of statistical secrecy of a protocol input. By proving that an information-theoretic definition is essentially equivalent to a definition based on the L_1-norm, which generalizes previous definitions in cryptology, we opened ways for more elegant proofs of secrecy, because information is a measure that allows many types of combinations. We demonstrated this in the proof of a multi-bit commitment scheme and a proof of the composability of protocols.

The commitment scheme is of independent interest because it is efficient and can be built from arbitrary collision-intractable hash functions, which is both a weak assumption in theory and useful in practice, since very efficient hash functions exist. An interesting open question is whether the reverse implication is also true, i.e., whether the existence of efficient multi-bit commitment schemes implies the existence of collision-intractable hash functions.

Acknowledgments

We thank Flemming Topsø for pointing us to Pinsker's inequality, Joachim Biskup for helpful remarks, and Joe Kilian for suggesting the advantage-based definition of statistical secrecy.

References

[1] C. H. Bennett, G. Brassard, C. Crépeau, and U. Maurer, "Generalized privacy amplification," *IEEE Trans. Inform. Theory*, vol. 41, no. 6, pp. 1915-1923, 1995.

[2] G. Bleumer, B. Pfitzmann, and M. Waidner, "A remark on a signature scheme where forgery can be proved," in *Advances in Cryptology - Proc. EUROCRYPT '90*, Berlin: Springer-Verlag, 1991, pp. 441–445.

[3] J. Bos, D. Chaum, and G. Purdy, "A voting scheme," unpublished manuscript, presented at the rump session of *CRYPTO '88*.

[4] J. Boyar, S. A. Kurtz, and M. W. Krentel, "A discrete logarithm implementation of perfect zero-knowledge blobs," *J. Cryptology*, vol. 2, no. 2, pp. 63–76, 1990.

[5] G. Brassard, D. Chaum, and C. Crépeau, "Minimum Disclosure Proofs of Knowledge," *J. Computer and System Sciences*, vol. 37, pp. 156–189, 1988.

[6] G. Brassard and C. Crépeau, "Nontransitive transfer of confidence: a perfect zero-knowledge interactive protocol for SAT and beyond," in *Proc. 27th IEEE Symp. Foundations of Computer Science*, 1986, pp. 188–195.

[7] J. L. Carter and M. N. Wegman, "Universal classes of hash functions," *J. Computer and System Sciences*, vol. 18, pp. 143–154, 1979.

[8] D. Chaum, I. Damgård, and J. van de Graaf, "Multiparty computations ensuring privacy of each party's input and correctness of the result," in *Advances in Cryptology - Proc. CRYPTO '87*, Berlin: Springer-Verlag, 1988, pp. 87–119.

[9] D. Chaum, E. v. Heijst, and B. Pfitzmann, "Cryptographically strong undeniable signatures, unconditionally secure for the signer," in *Advances in Cryptology - Proc. CRYPTO '91*, Berlin: Springer-Verlag, 1992, pp. 470–484.

[10] R. Cramer and I. Damgård, *Linear Zero-Knowledge - A Note on Efficient Zero-Knowledge Proofs and Arguments*, BRICS report series, RS-96-7, Aarhus University, Dept. of Computer Science.

[11] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems*, New York: Academic Press, 1981.

[12] I. Damgård, "Collision free hash functions and public key signature schemes," in *Advances in Cryptology - Proc. EUROCRYPT '87*, Berlin: Springer-Verlag, 1988, pp. 203–216.

[13] I. Damgård, "A design principle for hash functions," in *Advances in Cryptology - Proc. CRYPTO '89*, Berlin: Springer-Verlag, 1990, pp. 416–427.

[14] H. Dobbertin, A. Bosselaers, and B. Preneel, "RIPEMD-160: A strengthened version of RIPEMD," in *Proc. 3rd Fast Software Encryption Workshop*, Berlin: Springer-Verlag, 1996, pp. 71-82.

[15] R. G. Gallager, *Information Theory and Reliable Communication.* New York: Wiley, 1968.

[16] S. Goldwasser, S. Micali, "Probabilistic encryption," *J. Computer and System Sciences*, vol. 28, pp. 270–299, 1984.

[17] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof systems," *SIAM J. Computing*, vol. 18, no. 1, pp. 186–208, 1989.

[18] J. Kilian, "A Note on Efficient Proofs and Arguments," in *Proc. 24th Annual ACM Symp. Theory of Computing*, 1992, pp. ?–?.

[19] J. Kilian, "Efficient Interactive Arguments," in *Advances in Cryptology - Proc. CRYPTO '95*, Berlin: Springer-Verlag, 1995, pp. 311–324.

[20] J. Kilian, S. Micali, and R. Ostrovsky, "Minimum resource zero-knowledge proofs," in *Proc. 30th IEEE Symp. Foundations of Computer Science*, 1989, pp. 474–479.

[21] S. Micali, C. Rackoff, and B. Sloan, "The notion of security for probabilistic cryptosystems," *SIAM J. Computing*, vol. 17, no. 2, pp. 412–426, 1988.

[22] M. Naor, "Bit commitment using randomness," *J. Cryptology*, vol. 4, no. 2, pp. 151–158, 1991.

[23] M. Naor, R. Ostrovsky, R. Venkatesan, and M. Yung, "Zero-knowledge arguments for NP can be based on general complexity assumptions," in *Advances in Cryptology - Proc. CRYPTO '92*, Berlin: Springer-Verlag, 1993, pp. 196–214.

[24] M. Naor and M. Yung, "Universal one-way hash functions and their cryptographic applications," in *Proc. 21st Annual ACM Symp. Theory of Computing*, 1989, pp. 33–43.

[25] Y. Oren, "On the cunning power of cheating verifiers: some observations about zero-knowledge proofs," in *Proc. 28th IEEE Symp. Foundations of Computer Science*, 1987, pp. 462–471.

[26] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Advances in Cryptology - Proc. CRYPTO '91*, Berlin: Springer-Verlag, 1992, pp. 129–140.

[27] *Secure Hash Standard*, Federal Information Processing Standards Publication FIPS PUB 180-1, 1995.

[28] M. Tompa and H. Woll, "Random self-reducibility and zero knowledge proofs of possession of information," in *Proc. 28th IEEE Symp. Foundations of Computer Science*, 1987, pp. 472–482.

[29] A. C. Yao, "Theory and applications of trapdoor functions," in *Proc. 23rd IEEE Symp. Foundations of Computer Science*, 1982, pp. 80–91.

Recent Publications in the BRICS Report Series

RS-96-45 Ivan B. Damgård, Torben P. Pedersen, and Birgit Pfitzmann. *Statistical Secrecy and Multi-Bit Commitments*. November 1996. 30 pp.

RS-96-44 Glynn Winskel. *A Presheaf Semantics of Value-Passing Processes*. November 1996. 23 pp. Extended and revised version of paper appearing in Montanari and Sassone, editors, *Concurrency Theory: 7th International Conference*, CONCUR '96 Proceedings, LNCS 1119, 1996, pages 98–114.

RS-96-43 Anna Ingólfsdóttir. *Weak Semantics Based on Lighted Button Pressing Experiments: An Alternative Characterization of the Readiness Semantics*. November 1996. 36 pp. An extended abstract to appear in the proceedings of the *10th Annual International Conference of the European Association for Computer Science Logic*, CSL '96.

RS-96-42 Gerth Stølting Brodal and Sven Skyum. *The Complexity of Computing the k-ary Composition of a Binary Associative Operator*. November 1996. 15 pp.

RS-96-41 Stefan Dziembowski. *The Fixpoint Bounded-Variable Queries are PSPACE-Complete*. November 1996. 16 pp. Presented at the *10th Annual International Conference of the European Association for Computer Science Logic*, CSL '96.

RS-96-40 Gerth Stølting Brodal, Shiva Chaudhuri, and Jaikumar Radhakrishnan. *The Randomized Complexity of Maintaining the Minimum*. November 1996. 20 pp. To appear in a special issue of *Nordic Journal of Computing* devoted to the proceedings of SWAT '96. Appears in Karlson and Lingas, editors, *Algorithm Theory: 5th Scandinavian Workshop*, SWAT '96 Proceedings, LNCS 1097, 1996, pages 4–15.

RS-96-39 Hans Hüttel and Sandeep Shukla. *On the Complexity of Deciding Behavioural Equivalences and Preorders – A Survey*. October 1996. 36 pp.