# Key Agreement from Weak Bit Agreement

Thomas Holenstein
Department of Computer Science
Swiss Federal Institute of Technology (ETH)
Zurich, Switzerland
thomahol@inf.ethz.ch

## ABSTRACT

Assume that Alice and Bob, given an authentic channel, have a protocol where they end up with a bit $S_A$ and $S_B$, respectively, such that with probability $\frac{1+\varepsilon}{2}$ these bits are equal. Further assume that conditioned on the event $S_A = S_B$ no polynomial time bounded algorithm can predict the bit better than with probability $1 - \frac{\delta}{2}$. Is it possible to obtain key agreement from such a primitive? We show that for constant $\delta$ and $\varepsilon$ the answer is yes if and only if $\delta > \frac{1-\varepsilon}{1+\varepsilon}$, both for uniform and non-uniform adversaries.

The main computational technique used in this paper is a strengthening of Impagliazzo's hard-core lemma to the uniform case and to a set size parameter which is tight (i.e., twice the original size). This may be of independent interest.

## Categories and Subject Descriptors

F.2.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity—*Nonnumerical Algorithms and Problems*; E.3 [**Data**]: Data Encryption

## General Terms

Algorithms, Theory

## Keywords

Cryptography, Hard-core Sets, Key Agreement

## 1. INTRODUCTION

Key agreement, introduced by Diffie and Hellman in their seminal paper [3] is a protocol for two parties Alice and Bob which can communicate over an authentic channel such that they end up with a common string $K$. Furthermore no efficient algorithm can find $K$, given only the communication. Given the fact that the security of such a protocol implies P $\neq$ NP, no such protocol has been proven secure, but instead the security is based on some assumption.

In cryptography, much study has been devoted to find relations between different such assumptions and primitives. For example, Impagliazzo and Luby show in [9] that implementations of essentially all non-trivial cryptographic tasks imply the existence of one-way functions. On the other hand, many important primitives can be realized if one-way functions exist. Examples include pseudorandom generators [7], pseudorandom functions [5], and pseudorandom permutations [12].

For key agreement no such reduction to one-way functions is known. In fact, in [10] it is shown that such a reduction must be inherently non-relativizing, and thus it seems very hard to find such a construction.

It is thus natural to ask whether key agreement can be based on a primitive which is seemingly weaker. An example of such a primitive is weak key agreement: Alice and Bob have a protocol such that they end up with some strings which are equal only with some reasonable (noticeable) probability. Furthermore the outcome might only be secret in some cases, similar to a weak one-way function. The most natural case is when the string is just a single bit, and the question considered in this paper is whether such a primitive is sufficient to achieve key agreement.

### 1.1 Previous Work

#### 1.1.1 Computational Key Agreement

Key agreement was introduced by Diffie and Hellman in [3] and a scheme based on an algebraic hardness problem was proposed. Other proposed schemes for public key encryption (which also achieve key agreement) are also based on specific assumptions, for example the schemes given in [17, 15]. In [10], Impagliazzo and Rudich show that it is not possible to base key agreement on one-way functions unless non-relativizing techniques are used. Analogously, in [18], Rudich proves that the number of rounds of a given key agreement protocol can not be reduced with a relativizing technique.

In [4] Dwork, Naor and Reingold study the question when a non-perfect public key cryptosystem can be improved to get a nearly perfect one. They also consider the variant where the given, non-perfect system encrypts single bits. Let the probability that the receiver decrypts a single bit correctly be $\frac{1+\varepsilon}{2}$, and assume that the probability that an efficient algorithm can predict the bit correctly given only the encryption can be bounded by $1 - \frac{\delta}{2}$. Dwork et al. show that for some universal constant $c > 0$, such a cryptosystem can be used to get a public-key cryptosystem if $\varepsilon$ is noticeable and $\delta \geq 1 - c\varepsilon^2$.

*STOC'05,* May 22-24, 2005, Baltimore, Maryland, USA.

### *1.1.2 Hard-core Results*

A key building block of our work is a variant of Impagliazzo's hard-core lemma, given in [8]. In [11] Klivans and Servedio gave a connection of this lemma to boosting algorithms in computational learning theory, and showed that Impagliazzo's algorithm gives in fact a boosting algorithm (cf. [19]). They also note that boosting algorithms are uniform constructions, which was a motivation to find a uniform version of Impagliazzo's Lemma. In [20] Trevisan gives another version of the hard-core lemma which can be applied in the uniform setting. The main difference between Lemma 2.5 and Trevisan's version is that our version is *not* applicable if the predicate is not samplable, i.e., we need an algorithm which efficiently generates pairs $(x, P(x))$. On the other hand, if such samples can be efficiently obtained our version can be significantly stronger – it can in fact happen that applying our version is interesting while Trevisan's version does not give any non-trivial result.

### *1.1.3 Information Theoretic Secret-Key Agreement*

The question whether key agreement is possible in an information theoretic sense if the players Alice, Bob and Eve have a large supply of random variables $X$, $Y$, and $Z$, respectively, which are distributed according to some fixed distribution $P_{XYZ}$ was posed in [13]. Also, for any such distribution the secret-key rate was defined. Intrinsic information, which gives an upper bound on the secret-key rate, was defined in [14], see also [1]. In [16], information of formation was defined, which will be used implicitly to give impossibility results.

## 1.2 Notation and Definitions

A function $\alpha : \mathbb{N} \rightarrow [0,1]$ is negligible if $\alpha \in o(n^c)$ for all $c < 0$, otherwise it is non-negligible. It is noticeable if $\alpha \in \Omega(n^c)$ for some $c < 0$.

We define a protocol by two Turing machines, which have some common communication tapes.

Definition 1.1. *A* protocol *is a pair of Turing machines $A$ and $B$, called Alice and Bob. Both machines have a read only input tape, a read only random tape, a work tape and two common unerasable communication tapes $\Gamma_A$ and $\Gamma_B$. The machine $A$ writes one symbol on $\Gamma_A$ in every step, and $B$ writes one symbol on $\Gamma_B$ in every step.*

*In an execution of the protocol the machines $A$ and $B$ do alternating steps. The communication $\Gamma$ of a protocol denotes the contents of the communication tapes after a run. With $S_A$ we denote the content of the work tape of $A$ after the run, and $S_B$ is the content of the work tape of $B$ after the run.*

In the following we define the computational security of such a protocol. In this context, $\delta$ and $\varepsilon$ can be functions of $n$. We assume that they are computable in time polynomial in $n$. Reasonable values for $\delta$ and $\varepsilon$ are in the range $[0,1]$. Furthermore, we only consider the case where $S_A$ and $S_B$ are bits, and the protocol is symmetric, i.e., $\Pr[S_A = S_B = 0] = \Pr[S_A = S_B = 1]$, and $\Pr[S_A = 0] = \Pr[S_B = 0] = \frac{1}{2}$. It is easy to see that any protocol which yields bits can be modified to be symmetric.

Definition 1.2. *A* $\delta$-secure secret bit agreement (SBA) protocol *is a protocol where $A$ and $B$ get input $1^n$, such that $S_A, S_B \in \{0,1\}$, $\Pr[S_A = 0] = \Pr[S_B = 0] = \frac{1}{2}$, and for any polynomial time Turing machine $E$ for all but finitely many $n$ the inequality $\Pr[E(1^n, C) = S_A \mid S_A = S_B] < 1 - \frac{\delta}{2}$ holds. The protocol has correlation $\varepsilon$, if $\Pr[S_A = S_B = 1] = \Pr[S_A = S_B = 0] \geq \frac{1+\varepsilon}{4}$.*

For key agreement, both $\varepsilon$ and $\delta$ must be close to 1.

Definition 1.3. *A $\delta$-secure SBA protocol with correlation $\varepsilon$ achieves key agreement if $1-\delta$ and $1-\varepsilon$ are negligible in $n$.*

The goal of this paper is to answer the question whether a $\delta$-secure SBA protocol with correlation $\varepsilon$ implies key agreement.

## 1.3 Information Theoretic Setting

Consider a $\delta$-secure SBA protocol with correlation $\varepsilon$. Intuitively, this should be similar to the scenario where some oracle distributes random variables $X$, $Y$ and $Z$ to Alice, Bob and Eve, respectively. In this case $X$ and $Y$ will be over the alphabet $\mathcal{X} = \mathcal{Y} = \{0,1\}$, and the joint distribution of $X$ and $Y$ is defined by $\Pr[X = 0] = \Pr[Y = 0] = \frac{1}{2}$ and $\Pr[X = Y = 1] = \Pr[X = Y = 0] \geq \frac{1+\varepsilon}{4}$. Furthermore, if $X = Y$ Eve gets information about $X$ and $Y$ only with probability $1 - \delta$. The notion of $(\varepsilon, \delta)$-secure random variables formalizes this intuition. In the following $I(X;Y)$ is the mutual information (cf. [2]), and for an event $\mathcal{E}$ we define the conditional mutual information $I(X;Z \mid \mathcal{E})$ as the mutual information of the distribution of $X$ and $Z$ conditioned on $\mathcal{E}$.

Definition 1.4. *For any set $\mathcal{Z}$, a triple $X \times Y \times Z$ of random variables over $\{0,1\} \times \{0,1\} \times \mathcal{Z}$ is $(\varepsilon, \delta)$-secure if*

- $\Pr[X = 0] = \Pr[X = 1] = \Pr[Y = 0] = \Pr[Y = 1] = \frac{1}{2}$.
- $\Pr[X = Y = 0] = \Pr[X = Y = 1] \geq \frac{1+\varepsilon}{4}$.
- *There exists an event $\mathcal{E}$ which implies $X = Y$ such that $\Pr[\mathcal{E} \mid X = Y] \geq \delta$ and $I(X;Z \mid \mathcal{E}) = 0$.*

We consider protocols where the input tape of $A$ and $B$ is filled with infinitely many instantiations of $X$ and $Y$ from $(\varepsilon, \delta)$-secure random variables. Given this, we aim for a protocol where Alice and Bob end with the same bit, such that given the communication and all instances of $Z$, the bit is (information theoretically) indistinguishable from a random bit.

Definition 1.5. *Let $\varepsilon : \mathbb{N} \rightarrow [0,1]$ and $\delta : \mathbb{N} \rightarrow [0,1]$ be given. Let $(X_i, Y_i, Z_i)_{i \in \mathbb{N}}$ be independent $(\varepsilon, \delta)$-secure random variables. Let $X = (X_i)_{i \in \mathbb{N}}$, $Y = (Y_i)_{i \in \mathbb{N}}$ and $Z = (Z_i)_{i \in \mathbb{N}}$.*

*An information theoretic secret-key agreement protocol for $(\varepsilon(n), \delta(n))$-secure random variables is a protocol, where Alice gets input $(n, X)$, Bob gets input $(n, Y)$, and for the respective outputs $S_A$ and $S_B$ and communication $\Gamma$ of the protocol, $(S_A, S_B, \Gamma Z)$ is $(1 - 2^{-n}, 1 - 2^{-n})$-secure.*

*The protocol is* efficient *if the running time for both Alice and Bob is in* $\mathrm{poly}(n)$.

In Section 2, we will show that we can combine an information theoretic secret-key agreement protocol for $(\varepsilon, \delta)$-secure random variables as defined above with a $\delta$-secure SBA protocol that has correlation $\varepsilon$. Namely, every time the information theoretic protocol requests a random variable, we run the SBA protocol and use it as such an instance. This is also possible for non-constant $\delta$ and $\varepsilon$.

In Section 3 we show that for constants $\delta$ and $\varepsilon$ an efficient information theoretic protocol exists if $\delta > \frac{1-\varepsilon}{1+\varepsilon}$. The resulting protocol does not require any secrecy in case $X \neq Y$. On the other hand, if $\delta \leq \frac{1-\varepsilon}{1+\varepsilon}$, a $\delta$-secure SBA protocol with correlation $\varepsilon$ can be achieved without any computational hardness, which even satisfies that in case $X \neq Y$ the bits are information theoretically secure.

## 2. COMPUTATIONAL ASPECTS

The key lemma to show that efficient information theoretic protocols can be used computationally as well is a variant of Impagliazzo's hard-core lemma. Impagliazzo's lemma states that every predicate which is mildly hard on average for circuits of some size $s$ (no circuit can predict the predicate with probability higher than $1-\delta$ for some $\delta > 0$) has a large set of inputs (fraction $\delta$) such that on these inputs the predicate is very hard for all circuits of some smaller size $s'$.

We will use a modified version of the lemma, which has two improvements over the previously known versions: in Section 2.1 we show that the size of the hard-core set can be made twice as big[1], which is tight. The proof given for this is very similar to Nisan's proof in [8], see also [6]. In Section 2.2 we generalize the result to the uniform setting, this time in a way very similar to Impagliazzo's proof.

Finally, in Section 2.3 we show how the lemma implies that information theoretic protocols can be used in the computational setting.

### 2.1 Non-Uniform Hard-Core Sets

Assume that Alice and Bob have a protocol for bit agreement such that they always agree ($\varepsilon = 1$) on the outcome. Further assume that Eve, given the communication only, cannot always predict what Alice and Bob agreed on, but only with probability $1 - \frac{\delta}{2}$. Intuitively, one expects that for some of the possible randomness Alice and Bob use (about fraction $\delta$), it will be very difficult for Eve to find the bit Alice and Bob agreed on, while for others it might be rather easy. The following lemma – a variant of Impagliazzo's hard-core lemma – formalizes this intuition for non-uniform adversaries: $x \in \mathcal{R}_n \subseteq \{0,1\}^n$ will be the concatenated randomness of Alice and Bob, $f(x)$ the bit Alice and Bob agree on, $g(x)$ the communication between Alice and Bob, and $C'$ a circuit for Eve which is supposed to predict the predicate. We use the subset $\mathcal{R}_n \subseteq \{0,1\}^n$ of possible randomness because in general Alice and Bob will not always agree on the output ($\varepsilon < 1$), and $\mathcal{R}_n$ will only consist of those possible random strings for which the output of Alice and Bob will be equal (note that for our application $|\mathcal{R}_n| \geq 2^n/2$ trivially holds, because we assume that Alice and Bob agree in more than half of the cases).

To reduce the number of parameters, we assume that $g : \mathcal{R}_n \to \{0,1\}^n$ keeps the number of bits constant. Note that padding can be used if that is not the case. In the following lemma, reasonable values for $\delta$ and $\gamma$ are in the range $[0,1]$. The upper bound to the size of the circuits in the following lemma is not important in our application because every function $f : \{0,1\}^n \to \{0,1\}$ can be computed by circuits of size $\mathcal{O}(2^n/n)$ (see [22]), and thus we will only apply this lemma to circuits of significantly smaller size.

LEMMA 2.1 (NON-UNIFORM HARD-CORE LEMMA). *Let $\mathcal{R}_n \subseteq \{0,1\}^n$ be a set with $|\mathcal{R}_n| \geq 2^{n-1}$, $f : \mathcal{R}_n \to \{0,1\}$ be any predicate, $g : \mathcal{R}_n \to \{0,1\}^n$ any function, and $\gamma$, $\delta$ constants.*

*If for any constant $s' \leq 2^n \frac{\delta^2}{32}$ for all circuits $C'$ of size $s'$*

$$\Pr_{x \leftarrow \mathcal{R}_n}\big[C'\big(g(x)\big) = f(x)\big] \leq 1 - \frac{\delta}{2}, \qquad (1)$$

*then there exists a set $\mathcal{S} \subseteq \mathcal{R}_n$ with size $|\mathcal{S}| \geq \delta|\mathcal{R}_n|$ such that for all circuits $C$ of size $s = \frac{\gamma^2}{32n}s' - \mathcal{O}(1)$:*

$$\Pr_{x \leftarrow \mathcal{S}}[C(g(x)) = f(x)] < \frac{1+\gamma}{2}. \qquad (2)$$

Lemma 2.1 is proven by contradiction. The assumption that for every set $\mathcal{S}$ there exists a circuit $C$ of size $s$ which contradicts (2) is used to get a circuit $C'$ which contradicts (1). This reduction is done in three steps. In Lemma 2.2 we show that if for every set $\mathcal{S}$ of size $\delta|\mathcal{R}_n|$ there is a circuit which is correct with probability at least $\frac{1+\gamma}{2}$, then also for every distribution over $\mathcal{R}_n$ with min-entropy $\log(\delta|\mathcal{R}_n|)$ a circuit which has very similar advantage exists. In Lemma 2.3 we show that this implies that there exists a small collection of circuits with positive advantage on *every* set $\mathcal{S}$ of size $\delta|\mathcal{R}_n|$ (note the change in quantifiers: the collection is now the same for every set). Lemma 2.4 shows that such a collection is sufficient to obtain a circuit which contradicts the assumption of Lemma 2.1. It is the only new part of the proof and enables us to strengthen the hard-core lemma to twice the size (the rest can be found in [8]).

#### 2.1.1 Sets to Measures

We first show that the existence of circuits which perform well on sets implies circuits performing well on distributions with high min-entropy as well. Instead of distributions with high min-entropy we consider measures $M : \mathcal{R}_n \to [0,1]$. The *density* of a measure $M$ is $\mu(M) := \left|\mathcal{R}_n\right|^{-1} \sum_{x \in \mathcal{R}_n} M(x)$. An element $x$ is chosen according to $M$ if the probability of $x$ being chosen is proportional to $M(x)$. Note that the min-entropy of the distribution induced by a measure with density $\delta$ is at least $\log(\delta|\mathcal{R}_n|)$. A set $\mathcal{S}$ is chosen according to $M$ if every element $x$ is in the set independently of the other elements with probability $M(x)$.

The following lemma, converting circuits for sets to circuits for measures, is in fact not necessary for the applications we have in mind. One could formulate Lemma 2.1 using measures instead of sets. However, in applications it is often more intuitive to deal with sets instead of measures.

The lemma states that for a fixed measure $M$ with density $\delta$, if we choose a set $\mathcal{S}$ according to this measure, with overwhelming probability all circuits of some limited size perform the same on $\mathcal{S}$ as they do on $M$. The lemma is slightly stronger than what is needed here, but it will be needed in this form in Section 2.2.

LEMMA 2.2. *Let $\mathcal{R}_n$, $f$ and $g$ be as in Lemma 2.1, $\frac{1}{2} > \gamma$, $\delta > 0$ be fixed and $M : \mathcal{R}_n \to [0,1]$ be any measure with density $\mu(M) \geq \delta$. The probability that for a random set $\mathcal{S}$ chosen according to $M$ there exists a circuit $C$ with* $\mathrm{Size}(C) \leq$

[1] Actually, the previously known variant can be bootstrapped in order to obtain a set size which is as big as ours up to a factor which is in $o(1)$. In any case, our version is more direct.

$2^n \frac{\gamma^2\delta^2}{64n}$ *satisfying*

$$\Big|\Pr_{x\leftarrow M}[C(g(x))=f(x)] - \Pr_{x\leftarrow \mathcal{S}}[C(g(x))=f(x)]\Big| \geq \gamma \quad (3)$$

*is less than* $2^{-2^n\gamma^2\delta^2/64}$.

Proof Sketch. Using the Hoeffding bound we can show that for any fixed circuit the deviation of advantage from the expectation is very small with high probability. Because there are not so many circuits which are smaller than the above bound, the union-bound implies the statement. □

### 2.1.2 A Collection of Circuits for Every Set

We now prove that if for every measure $M$ with $\mu(M) \geq \delta$ there exists a circuit which is good, this implies that there exists a small collection of circuits which perform well on *every* set $\mathcal{S}$ of size $\delta|\mathcal{R}_n|$.

Lemma 2.3. *Let $\mathcal{R}_n$, $f$ and $g$ be as in Lemma 2.1, and $\gamma, \delta > 0$. Assume that for every measure $M : \mathcal{R}_n \to [0,1]$ with density $\mu(M) \geq \delta$ there exists a circuit $C_M$ of size $s$ for which*

$$\Pr_{x\leftarrow M}[C_M(g(x))=f(x)] \geq \frac{1+\gamma}{2}$$

*Then there exists a collection $\mathcal{C}$ of $\frac{8n}{\gamma^2}$ circuits of size $s$ such that for every set $\mathcal{S}$ of size $|\mathcal{S}| \geq \delta|\mathcal{R}_n|$*

$$\Pr_{C\leftarrow\mathcal{C}, x\leftarrow\mathcal{S}}[C(g(x))=f(x)] > \frac{1}{2}.$$

Proof. Consider the following zero-sum game of two players Alice and Bob: Alice chooses a circuit $C$ of size at most $s$, and simultaneously Bob chooses a set $\mathcal{S} \subseteq \mathcal{R}_n$ with $|\mathcal{S}| \geq \delta|\mathcal{R}_n|$. The payoff for Alice is $\Pr_{x\leftarrow\mathcal{S}}[C(g(x))=f(x)]$.

A randomized strategy for Bob is a distribution on sets of size at least $\delta|\mathcal{R}_n|$, and corresponds to a distribution $M$ on $\mathcal{R}_n$ with $\mu(M) \geq \delta$. For any such strategy, the assumption of the lemma implies that Alice has a strategy to obtain a value of at least $\frac{1+\gamma}{2}$. According to von Neumann's minmax Theorem [21] this means that there exists a strategy (i.e., a distribution on circuits) for Alice, such that for no strategy of Bob the payoff is lower than $\frac{1+\gamma}{2}$. Thus, for this distribution $\mathcal{C}'$ we obtain for every set $S$ with $|S| \geq \delta|\mathcal{R}_n|$, that $\Pr_{C\leftarrow\mathcal{C}', x\leftarrow\mathcal{S}}[C(g(x))=f(x)] \geq \frac{1+\gamma}{2}$.

Using Chernoff's bound it is now easy to show that $\lambda = \frac{8n}{\gamma^2}$ circuits $C_1, \ldots, C_\lambda$ from $\mathcal{C}'$ exist for which, for every $x \in \mathcal{R}_n$, the fraction of circuits answering 1 deviates from the expected value by less than $\gamma/2$. This collection $\mathcal{C}$ must satisfy the statement of the lemma. □

### 2.1.3 Combining the Circuits in the Collection

The key observation to improve over [8] in the set size is given in the following lemma, which states that to do so, a *collection* of circuits which does well on average for every set is sufficient. The proof uses a trick very similar as one used by Levin in order to proof the XOR-Lemma, see [6], namely it does a randomized decision instead of taking the majority.

Lemma 2.4. *Let $\mathcal{R}_n$, $f$ and $g$ be as in Lemma 2.1, $\delta > 0$ be fixed and $\mathcal{C}$ a collection of circuits such that for every $\mathcal{S} \subseteq \mathcal{R}_n$ of size $|\mathcal{S}| \geq \delta|\mathcal{R}_n|$*

$$\Pr_{C\leftarrow\mathcal{C}, x\leftarrow\mathcal{S}}[C(g(x))=f(x)] > \frac{1}{2}.$$

*Then there is a circuit $C'$ of size $\mathcal{O}(|\mathcal{C}|) + \sum_{C\in\mathcal{C}} \mathrm{Size}(C)$ such that $\Pr[C'(g(x))=f(x)] > 1-\frac{\delta}{2}$.*

Proof. Let $\alpha_{\mathrm{corr}}(x) = 2\Pr_{C\leftarrow\mathcal{C}}[C(g(x))=f(x)] - 1$ be the expected advantage of a circuit from $\mathcal{C}$ on $x$. Analogous, let $\alpha_1 = 2\Pr_{C\leftarrow\mathcal{C}}[C(g(x))=1] - 1$. Consider a subset $\mathcal{S} \subseteq \mathcal{R}_n$ of size $\delta|\mathcal{R}_n|$ for which the sum $\sum_{x\in\mathcal{S}} \alpha_{\mathrm{corr}}(x)$ is minimal, and let $\varphi > 0$ be the maximum of $\alpha_{\mathrm{corr}}(x)$, $x \in \mathcal{S}$.

We first describe a randomized circuit. On input $g(x)$, circuit $C'$ first evaluates all circuits in the collection $\mathcal{C}$ and then finds $\alpha_1(x)$. It then outputs 1 with probability

$$\Pr[C'(g(x))=1] = \begin{cases} 0 & \text{if } \alpha_1(x) \leq -\varphi \\ \frac{1}{2} + \frac{\alpha_1(x)}{2\varphi} & \text{if } -\varphi < \alpha_1(x) < \varphi \\ 1 & \text{if } \varphi \leq \alpha_1(x). \end{cases}$$

The probability that $C'(g(x))$ equals $f(x)$ is $\frac{1}{2} + \frac{\alpha_{\mathrm{corr}}(x)}{2\varphi}$, truncated at 0 and 1. Therefore for $x \notin \mathcal{S}$, the circuit will always be correct.

On the other hand, since $\mathcal{S}$ has size $\delta|\mathcal{R}_n|$, the assumption of the lemma implies $\Pr_{C\leftarrow\mathcal{C}, x\leftarrow\mathcal{S}}[C(g(x))=f(x)] > \frac{1}{2}$, and thus $\mathrm{E}_{x\leftarrow\mathcal{S}}[\alpha_{\mathrm{corr}}(x)] > 0$. For a fixed $x \in \mathcal{S}$ we obtain

$$\Pr[C'(g(x))=f(x)] = \max\Big(0, \frac{1}{2} + \frac{\alpha_{\mathrm{corr}}(x)}{2\varphi}\Big) \geq \frac{1}{2} + \frac{\alpha_{\mathrm{corr}}(x)}{2\varphi}$$

and thus $\Pr_{x\leftarrow\mathcal{S}}[C'(g(x))=f(x)] > \frac{1}{2}$, which together with the above statement implies $\Pr[C'(g(x))=f(x)] > 1-\frac{\delta}{2}$.

Note that the total size of $C'$ is the sum of the sizes of the circuit in $\mathcal{C}$ plus the size needed to compute the output. The output bits of the circuits in $\mathcal{C}$ can now be sorted with linear complexity (see [22, Chapter 3.4]). The randomness used now just selects one of the output bits with a certain probability distribution. We can fix the randomness to the value for which the circuit has highest probability in predicting $f(x)$ overall, and thus obtain the lemma. □

### 2.1.4 Assembling the Parts

We can use Lemmata 2.2, 2.3 and 2.4 to proof Lemma 2.1.

Proof (of Lemma 2.1). Assume for a contradiction, that for every set $\mathcal{S}$ of size $|\mathcal{S}| \geq \delta|\mathcal{R}_n|$ there exists a circuit $C$ of size $s$ such that

$$\Pr_{x\leftarrow\mathcal{S}}[C(g(x))=f(x)] \geq \frac{1+\gamma}{2}.$$

The assumption $s' \leq 2^n \frac{\delta^2}{32}$ of the lemma implies $s \leq 2^n \frac{\delta^2\gamma^2}{1024n}$. Using Lemma 2.2 this implies that for every measure $M$ with density $\delta$ there exists a circuit $C''$ of size $s$ satisfying

$$\Pr_{x\leftarrow\mathcal{S}}[C''(g(x))=f(x)] \geq \frac{1+\gamma/2}{2}.$$

Lemma 2.3 then implies that there exists a collection $\mathcal{C}$ of $32n\gamma^{-2}$ circuits of size $s$ with

$$\Pr_{C\leftarrow\mathcal{C}, x\leftarrow\mathcal{S}}[C(g(x))=f(x)] > \frac{1}{2}.$$

Lemma 2.4 states that we can combine these circuits to obtain one circuit $C'$ of size $\mathcal{O}(32n\gamma^{-2}) + 32ns\gamma^{-2}$ for which

$$\Pr[C'(g(x))=f(x)] > 1-\frac{\delta}{2},$$

and thus we have a contradiction. □

## 2.2 Uniform Hard-Core Sets

Lemma 2.1 is only applicable in the non-uniform settings, i.e., where Eve is modeled by non-uniform circuits. In this section we present a similar lemma for the uniform case. The main step for this is to assume that an *efficient algorithm* produces the circuits needed. In the following, let $\chi_S$ be the characteristic function of a set $S$, i.e., $\chi_S(x) = 1 \iff x \in S$ and $\chi_S(x) = 0$ otherwise.

LEMMA 2.5 (UNIFORM HARD-CORE LEMMA). *Let an efficiently samplable set $\mathcal{R}_n \subseteq \{0,1\}^n$ with $|\mathcal{R}_n| \geq 2^{n-1}$ be given. Let the functions $f : \mathcal{R}_n \to \{0,1\}$, $g : \mathcal{R}_n \to \{0,1\}^n$, $\delta : \mathbb{N} \to [0,1]$ and $\gamma : \mathbb{N} \to [0,1]$ be computable in time* poly$(n)$*, and assume that $\delta$ and $\gamma$ are noticeable.*

*Assume that there is no polynomial time algorithm $B$ such that*

$$\Pr_{x \leftarrow \mathcal{R}_n}[B(g(x)) = f(x)] \geq 1 - \frac{\delta}{2}$$

*for infinitely many $n$. Then, there is no polynomial time oracle algorithm $A^{(\cdot)}$ such that for infinitely many $n$ the following holds: for any set $\mathcal{S} \subseteq \mathcal{R}_n$ with $|\mathcal{S}| \geq \delta|\mathcal{R}_n|$, $A^{\chi_{\mathcal{S}}}$ outputs a circuit $C$ satisfying*

$$\mathrm{E}\big[\Pr_{x \leftarrow \mathcal{S}}[C(g(x)) = f(x)]\big] \geq \frac{1+\gamma}{2}$$

*(the expectation is over the randomness of $A$).*

As before, we need the notion of measures for the proof. Recall that a measure $M$ is a function $M : \mathcal{R}_n \to [0,1]$, the density of $M$ is defined as $\mu(M) := |\mathcal{R}_n|^{-1} \sum_x M(x)$ and a set $\mathcal{S}$ is chosen according to $M$ if $x \in \mathcal{S}$ independently of all other elements with probability $M(x)$.

The proof is by contradiction. The basic idea is to use the algorithm $A$ in the following way: we start with an empty collection[2] $\mathcal{C}$ of circuits, and add circuits one by one to $\mathcal{C}$. In every step, the collection $\mathcal{C}$ is used to define a measure $M$ with $\mu(M) \geq \delta$. The measure is then used to define a set with size at least $\delta|\mathcal{R}_n|$, and this set is used with $A$ to obtain another circuit, which is then added to the collection. This is repeated until for the collection either the majority of the circuits answers correctly on slightly more than fraction $1 - \frac{\delta}{2}$, or else for every set $\mathcal{S}$ of size $\delta|\mathcal{R}_n|$, a random circuit of $\mathcal{C}$ has probability slightly higher than $1/2$ of being correct on $\mathcal{S}$ (a similar condition as for Lemma 2.4). We then show that in both cases we can obtain a circuit $C'$ from $\mathcal{C}$ satisfying

$$\Pr[C'(g(x)) = f(x)] > 1 - \frac{\delta}{2},$$

which will be enough to get a contradiction.

### 2.2.1 The Idealized Algorithm

We first describe an idealized version of the algorithm. The idealized version assumes that some characteristics of a given collection $\mathcal{C}$ of circuits (for example the density of the measure $M_{\mathcal{C},s}$ defined by $C$, see below) can be estimated up to some error margin, but the probability of a larger error is zero. Furthermore it assumes that an efficient algorithm exists which, for any measure $M$ with $\mu(M) \geq \delta$, returns a circuit satisfying $\Pr_{x \leftarrow M}[C(g(x)) = f(x)] \geq (1+\gamma)/2$. We will show in Section 2.2.2 how to drop these assumptions.

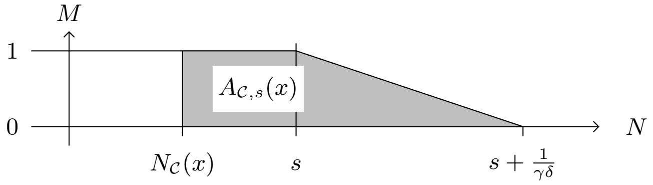

**Figure 1: The advantage $N_{\mathcal{C}}(x)$, measure $M_{\mathcal{C},s}(x)$, area $A_{\mathcal{C},s}(x)$ for one fixed $x$.**

For the collection $\mathcal{C}$ of circuits let

$$N_{\mathcal{C}}(x) := \big|\{C \in \mathcal{C} \mid C(g(x)) = f(x)\}\big| - \big|\{C \in \mathcal{C} \mid C(g(x)) \neq f(x)\}\big|.$$

The measure $M_{\mathcal{C},s}(x)$ used to request the next circuit depends on $N_{\mathcal{C}}$ and additionally on a number $s$ (which is initially 0 but will be increased while the collection is growing). It is defined as

$$M_{\mathcal{C},s}(x) := \begin{cases} 1 & N_{\mathcal{C}}(x) \leq s \\ 1 - (N_{\mathcal{C}}(x) - s)\gamma\delta & s < N_{\mathcal{C}}(x) < s + \frac{1}{\gamma\delta} \\ 0 & N_{\mathcal{C}}(x) \geq s + \frac{1}{\gamma\delta} \end{cases}$$

(cf. Figure 1).

In order to proof that our algorithm will stop we consider the area under the curve in Figure 1, starting from $N_{\mathcal{C}}(x)$. Formally, $A_{\mathcal{C},s}(x)$ is defined as

$$A_{\mathcal{C},s}(x) := \begin{cases} s - N_{\mathcal{C}(x)} + \frac{1}{2\gamma\delta} & N_{\mathcal{C}}(x) \leq s \\ 0 & N_{\mathcal{C}}(x) \geq s + \frac{1}{\gamma\delta} \\ \frac{M_{\mathcal{C},s}(x)}{2}\big(s + \frac{1}{\gamma\delta} - N_{\mathcal{C}}(x)\big) & \text{otherwise.} \end{cases} \tag{4}$$

The total area is also important, and thus let

$$A(\mathcal{C}, s) := \frac{1}{|\mathcal{R}_n|} \sum_{x \in \mathcal{R}_n} A_{\mathcal{C},s}(x).$$

The idealized version of the algorithm is shown in Figure 2. We use the following notation: The skip-statement does nothing. The semantics of the if statement as in

```
**if** $C_1 \to S_1$
   ▯ $C_2 \to S_2$
**fi**
```

for conditions $C_1$ and $C_2$, and statements $S_1$ and $S_2$ is that one condition which holds is chosen in an arbitrary way, and the corresponding statement is executed. It is important that we do not make any assumption which statement is executed if both conditions hold.

The algorithm given in Figure 2 is very simple: it adds the circuit which performs well on $M_{\mathcal{C},s}$ to the collection $\mathcal{C}$ as long as the measure has density at least $\delta$. If the density is too small, i.e., $\mu(M_{\mathcal{C},s}) \leq \delta$, then $s$ is increased before obtaining the circuit. This is repeated until the resulting collection is good enough to prove Lemma 2.5.

The if-statements in the idealized algorithm require sampling, and thus it is not possible to give an efficient implementation of the algorithm in Figure 2. On the other hand

[2] Formally, $\mathcal{C}$ is a multiset.

```
1  procedure GoodEnough(Collection $\mathcal{C}$):
2    $p := \min_{\mathcal{S}:|\mathcal{S}|\geq\delta|\mathcal{R}_n|} \Pr_{x\leftarrow\mathcal{S},C\leftarrow\mathcal{C}}[C(g(x)) = f(x)]$
3    $r := |\mathcal{R}_n|^{-1}|\{x \mid N_{\mathcal{C}}(x) \leq 0\}|$
4    if $p \geq \frac{1}{2} + \gamma\delta^2/32 \;\vee\; r \leq 7\delta/16 \rightarrow$ return true
5    ▯ $p \leq \frac{1}{2} + \gamma\delta^2/16 \;\wedge\; r \geq 3\delta/8 \rightarrow$ return false
6    fi
7  end GoodEnough.
8
9  procedure Imp$^+$:
10   $s := 0, \mathcal{C} := \emptyset$
11   while not GoodEnough($\mathcal{C}$) do
12       if $\mu(M_{\mathcal{C},s}) \leq \delta(1 + \gamma\delta/16) \rightarrow s := s + 1$
13       ▯ $\mu(M_{\mathcal{C},s}) \geq \delta \rightarrow$ skip
14       fi
15       $\mathcal{C} := \mathcal{C} \cup \{C_M\}$, where $C_M$ satisfies
16                $\Pr_{x\leftarrow M_{\mathcal{C},s}}[C_M(g(x)) = f(x)] > \frac{1+\gamma}{2}$.
17   od
18   return $\mathcal{C}$
19 end Imp$^+$.
```

**Figure 2: Algorithm for Proof of Lemma 2.5. The statement "skip" does nothing. In an if-statement, any line may be executed for which the guard evaluates to true.**

we can show that there exists an *efficient randomized implementation*, i.e., for any $\kappa \in \text{poly}(n)$ there exists an efficient algorithm which does exactly the same as the idealized version with probability at least $1 - 2^{-\kappa}$.

We show the correctness of the algorithm in several steps. First, we show that there exists an efficient randomized implementation for the loop. Then we show that in the idealized version the loop terminates after at most $4\gamma^{-2}\delta^{-3}$ iterations. Finally, we prove that a collection as returned by the idealized version is sufficient to prove Lemma 2.5.

### 2.2.2 Efficient Implementation of one Loop

To implement the algorithm in Figure 2, some knowledge about $\mu(M_{\mathcal{C},s})$, $p$ (in line 2) and $r$ (in line 3) is required. Also we need to make sure that we can obtain circuits for the measures as required in line 15. For a collection $\mathcal{C}$ of circuits, let $\text{Size}(\mathcal{C})$ be the sum of the sizes of the circuits in the collection.

Claim 2.6. *If* $\text{Size}(\mathcal{C}) \in \text{poly}(n)$ *then there exists an efficient randomized implementation of the conditional statements in lines 4 and 12.*

Proof. Using the Hoeffding bound it is clear that for any $\kappa \in \text{poly}(n)$ we can efficiently find samples $r'$ and $\mu'$ of $r$ and $\mu(M_{\mathcal{C},s})$, respectively, such that with probability $1-2^{-\kappa}$ both $|r' - r| < \delta/32$ and $|\mu' - \mu(M_{\mathcal{C},s})| < \gamma\delta^2/32$.

Furthermore, it is only slightly more involved to see that we can efficiently find an estimate $p'$ of $p$ such that with probability $1 - 2^{-\kappa}$ the estimate satisfies $|p' - p| < \gamma\delta^2/64$.

Making $\kappa \in \text{poly}(n)$ large enough this implies the claim. □

In order to show that line 15 can be implemented we first need to show that the measure satisfies $\mu(M_{\mathcal{C},s}) \geq \delta$.

Claim 2.7. *In every iteration, after line 14 the measure* $M_{\mathcal{C},s}$ *satisfies* $\mu(M_{\mathcal{C},s}) \geq \delta$.

Proof. The claim holds in the first round. Furthermore, the claim can only be wrong if $s$ is increased in line 12. In this case the measure cannot have decreased for any $x$ when compared with the iteration before. This implies that the total density is at least as big as one iteration earlier, which implies the claim by induction. □

Next, we show that an algorithm $A$ as in Lemma 2.2 (which we assume to exist for a contradiction) can be used to give a randomized implementation of line 15 of the idealized algorithm, which works for infinitely many $n$. For this, two modifications are necessary. First, instead of only producing circuits which are expected to perform well, we make sure it produces circuits which are nearly always performing well. Second, we use Lemma 2.2 to show that such an algorithm also works with measures instead of sets. In order to simplify the notation, we double the parameter $\gamma$ when compared with Lemma 2.5 (this does not make any difference in the statement of the lemma).

Claim 2.8. *Let* $\mathcal{R}_n$, $f$, $g$, $\delta$ *and* $\gamma$ *be as in Lemma 2.5. Let* $A^{(\cdot)}$ *be an polynomial time oracle algorithm such that for infinitely many* $n$*, for any set* $\mathcal{S} \subseteq \mathcal{R}_n$ *with* $|\mathcal{S}| \geq \delta|\mathcal{R}_n|$*,* $A^{\chi_\mathcal{S}}$ *outputs a circuit* $C$ *satisfying*

$$\mathrm{E}\big[\Pr_{x\leftarrow\mathcal{S}}[C(g(x)) = f(x)]\big] \geq \frac{1}{2} + \gamma.$$

*If* $\text{Size}(\mathcal{C}) \in \text{poly}(n)$*, then there exists an efficient randomized implementation of line 15 which is correct for infinitely many* $n$.

Proof. Note that, given an oracle for $M(x)$, it is possible to efficiently simulate an oracle $\chi_\mathcal{S}$ for a set $\mathcal{S}$ chosen according to $M$ (the answers must be cached). We thus run algorithm $A^{\chi_\mathcal{S}}$ for a set $\mathcal{S}$ chosen according to $M$. Because of Lemma 2.2 and Markov's inequality, the probability that a circuit $C$ is returned for which

$$q := \Pr_{x\leftarrow M}[C(g(x)) = f(x)] \geq \frac{1}{2} + \frac{7\gamma}{8}.$$

is noticeable (for infinitely many $n$).

Using Chernoff's inequality, for any $\kappa \in \text{poly}(n)$, the constructed algorithm finds an estimate $q'$ of $q$ such the probability that $|q' - q| > \frac{\gamma}{8}$ is at most $2^{-\kappa}$. In case the estimate is at least $\frac{1}{2} + \frac{3\gamma}{4}$, we return the circuit, otherwise we start over again with new samples.

Note that with probability $1 - 2^{-\kappa}$ a circuit is returned after polynomially many tries of the above. □

This implies that we can give an efficient randomized implementation of one loop of the algorithm.

Lemma 2.9. *Let* $\mathcal{R}_n$, $f$, $g$, $\delta$ *and* $\gamma$ *be as in Lemma 2.5. If there exists a polynomial time oracle algorithm* $A^{(\cdot)}$ *such that for infinitely many* $n$*, for any set* $\mathcal{S} \subseteq \mathcal{R}_n$ *with* $|\mathcal{S}| \geq \delta|\mathcal{R}_n|$*,* $A^{\chi_\mathcal{S}}$ *outputs a circuit* $C$ *satisfying*

$$\mathrm{E}[\Pr_{x\leftarrow\mathcal{S}}[C(g(x)) = f(x)]] \geq \frac{1}{2} + \gamma,$$

*and if* $\text{Size}(\mathcal{C}) \in \text{poly}(n)$*, then there exists an efficient randomized implementation of the loop of Algorithm Imp*$^+$ *in Figure 2 which is correct for infinitely many* $n$.

Proof. Claim 2.7 and 2.8 together imply that for infinitely many $n$ there exists an implementation of line 15 of the algorithm. Claim 2.6 shows that the if statements have a randomized implementation as well, which proves the lemma. □

### 2.2.3 Termination

We now show that the algorithm stops after at most $4\gamma^{-2}\delta^{-3}$ iterations. For this, we show that $A(\mathcal{C}, s) - \delta s$ decreases by at least $\gamma\delta^2/8$ in every iteration, and that the algorithm must stop if it gets smaller than 0. Note that initially $A(\emptyset, 0) = \frac{1}{2\gamma\delta}$. First, we show that adding a circuit to $\mathcal{C}$ (while leaving $s$ constant) decreases $A(\mathcal{C}, s)$ by at least $\frac{\gamma\delta}{2}$.

Claim 2.10. *If $C_M$ satisfies $\Pr_{x\leftarrow M_{\mathcal{C},s}}[C_M(g(x)) = f(x)] \geq \frac{1+\gamma}{2}$ and $\mu(M_{\mathcal{C},s}) \geq \delta$, then $A(\mathcal{C} \cup \{C_M\}, s) \leq A(\mathcal{C}, s) - \frac{\gamma\delta}{2}$.*

Proof. First, let $\mathcal{S}^+ := \{x \mid C_M(g(x)) = f(x)\}$ (i.e., the $x$ for which $C_M$ is correct) and $\mathcal{S}^- := \{x \mid C_M(g(x)) \neq f(x)\}$, and let $\mathcal{C}' = \mathcal{C} \cup \{C_M\}$.

Consider a fixed $x$. If $x \in S^+$, then $A_{\mathcal{C}',s}(x) \leq A_{\mathcal{C},s}(x) - M_{\mathcal{C},s}(x) + \gamma\delta/2$ (note that $N_{\mathcal{C}'}(x) = N_{\mathcal{C}}(x) + 1$, and with Figure 1 it is easy to see that the area decreases by at least $M_{\mathcal{C},s}(x)$ minus the small triangle which is cut off in case $N_{\mathcal{C}}(x)$ is such that $M_{\mathcal{C},s}(x)$ is not constant). Also, if $x \in S^-$ then $A_{\mathcal{C}',s}(x) \leq A_{\mathcal{C},s}(x) + M_{\mathcal{C},s}(x) + \gamma\delta/2$. Thus,

$$A(\mathcal{C}', s) \leq A(\mathcal{C}, s) + \frac{\gamma\delta}{2} + \frac{1}{|\mathcal{R}_n|}\Big(\sum_{x\in\mathcal{S}^-} M_{\mathcal{C},s}(x) - \sum_{x\in\mathcal{S}^+} M_{\mathcal{C},s}(x)\Big).$$

It is easy to see that $\Pr_{x\leftarrow M_{\mathcal{C},s}}[C_M(g(x)) = f(x)] \geq \frac{1+\gamma}{2}$ implies $\sum_{x\in\mathcal{S}^+} M_{\mathcal{C},s}(x) - \sum_{x\in\mathcal{S}^-} M_{\mathcal{C},s}(x) \geq \gamma \sum_x M_{\mathcal{C},s}(x)$, and using $\mu(M_{\mathcal{C},s}) \geq \delta$ we see that

$$A(\mathcal{C}', s) \leq A(\mathcal{C}, s) + \frac{\gamma\delta}{2} - \gamma\delta = A(\mathcal{C}, s) - \frac{\gamma\delta}{2}. \quad \square$$

Of course, if $s$ is increased in line 12, then the area $A(\mathcal{C}, s)$ will grow. We can give an upper bound on this:

Claim 2.11. *If $s$ is increased in line 12, then $A(\mathcal{C}, s+1) \leq A(\mathcal{C}, s) + \delta + \frac{\gamma\delta}{2} - \frac{\gamma\delta^2}{8}$.*

Proof. First we note that for any $x$, $A_{\mathcal{C},s+1}(x) \leq A_{\mathcal{C},s}(x) + M_{\mathcal{C},s}(x) + \gamma\delta/2$, and if $N_{\mathcal{C}}(x) \leq 0 \leq s$ then $A_{\mathcal{C},s+1}(x) \leq A_{\mathcal{C},s}(x) + M_{\mathcal{C},s}(x)$. Since the loop would have stopped if $\mathcal{S} := \{x \mid N_{\mathcal{C}}(x) \leq 0\}$ was smaller than $(3\delta/8)|\mathcal{R}_n|$, we get

$$\begin{aligned}A(\mathcal{C}, s+1) &\leq A(\mathcal{C}, s) + \frac{1}{|\mathcal{R}_n|}\Big(\sum_{x\in\mathcal{S}} M_{\mathcal{C},s}(x) + \sum_{x\notin\mathcal{S}}\Big(M_{\mathcal{C},s}(x) + \frac{\gamma\delta}{2}\Big)\Big)\\ &\leq A(\mathcal{C}, s) + \underbrace{\mu(M_{\mathcal{C},s})}_{\leq\delta(1+\gamma\delta/16)} + \Big(1 - \frac{3\delta}{8}\Big)\frac{\gamma\delta}{2}\\ &\leq A(\mathcal{C}, s) + \delta + \frac{\gamma\delta}{2} - \frac{\gamma\delta^2}{8}. \quad \square\end{aligned}$$

Claim 2.12. *In every iteration of the loop, $A(\mathcal{C}, s) - s\delta$ decreases by at least $\frac{\gamma\delta^2}{8}$.*

Proof. Combine Claim 2.10 and 2.11. □

Claim 2.13. *If $A(\mathcal{C}, s) - s\delta < 0$, then $\mathcal{C}$ is a collection which satisfies*

$$\Pr_{C\leftarrow\mathcal{C},x\leftarrow\mathcal{S}}[C(g(x)) = f(x)] > \frac{1}{2} + \frac{1}{4\gamma\delta|\mathcal{C}|}$$

*for every $\mathcal{S} \subseteq \mathcal{R}_n$ of size $|\mathcal{S}| \geq \delta|\mathcal{R}_n|$.*

Proof. Let $\mathcal{H} \subseteq \mathcal{R}_n$ be a set of size $\delta|\mathcal{R}_n|$ for which $\Pr_{C\leftarrow\mathcal{C},x\leftarrow\mathcal{H}}[C(g(x)) = f(x)]$ is minimized. Since

$$\Pr_{C\leftarrow\mathcal{C},x\leftarrow\mathcal{H}}[C(g(x)) = f(x)] = \frac{1}{2} + \frac{\sum_{x\in\mathcal{H}} N_{\mathcal{C}}(x)}{2|\mathcal{C}||\mathcal{H}|}$$

it is enough to show that $\sum_{x\in\mathcal{H}} N_{\mathcal{C}}(x) > \frac{|\mathcal{R}_n|}{2\gamma}$. From (4) we see that $A_{\mathcal{C},s}(x) \geq \frac{1}{2\gamma\delta} + s - N_{\mathcal{C}}(x)$, and this implies

$$\begin{aligned}\sum_{x\in\mathcal{H}} N_{\mathcal{C}}(x) &\geq \sum_{x\in\mathcal{H}} \frac{1}{2\gamma\delta} + s - A_{\mathcal{C},s}(x)\\ &\geq \delta|\mathcal{R}_n|\Big(\frac{1}{2\gamma\delta} + s\Big) - \sum_{x\in\mathcal{R}_n} A_{\mathcal{C},s}(x)\\ &= \frac{|\mathcal{R}_n|}{2\gamma} + \delta|\mathcal{R}_n|s - |\mathcal{R}_n|A(\mathcal{C}, s) > \frac{|\mathcal{R}_n|}{2\gamma}. \quad \square\end{aligned}$$

Lemma 2.14. *The loop of Algorithm Imp$^+$ is traversed at most $4\gamma^{-2}\delta^{-3}$ times.*

Proof. Initially the collection is empty, and thus $A(\mathcal{C}, s) = A(\emptyset, 0) = \frac{1}{2\gamma\delta}$. Since in every iteration $A(\mathcal{C}, s) - s\delta$ decreases by at least $\frac{\gamma\delta^2}{8}$, this means that after at most $4\gamma^{-2}\delta^{-3}$ iterations $A(\mathcal{C}, s) - s\delta < 0$, in which case Claim 2.13 implies that

$$\Pr_{C\leftarrow\mathcal{C},x\leftarrow\mathcal{S}}[C(g(x)) = f(x)] > \frac{1}{2} + \frac{\gamma\delta^2}{16}$$

(note that $|\mathcal{C}| \leq 4\gamma^{-2}\delta^{-3}$). Thus, the if statement in line 4 of the algorithm *must* return true (since the guard of line 5 is wrong), and the algorithm terminates. □

### 2.2.4 The Collection Yields a Circuit

Claim 2.15. *Let $\gamma$, $\delta$ be noticeable, $\mathcal{C}$ be a collection of circuits such that* $\text{Size}(\mathcal{C}) \in \text{poly}(n)$ *and for every set $\mathcal{S}$ of size $\delta|\mathcal{R}_n|$*

$$\Pr_{x\leftarrow\mathcal{S},C\leftarrow\mathcal{C}}[C(g(x)) = f(x)] > \frac{1}{2} + \frac{\gamma\delta^2}{16}$$

*then, there is a randomized circuit $C'$ of size* $\text{poly}(n)$ *for which*

$$\Pr[C'(g(x)) = f(x)] > 1 - \frac{\delta}{2} + \frac{\gamma^2\delta^4}{2048}$$

*Furthermore, for any $\kappa \in$* $\text{poly}(n)$ *such a circuit $C'$ can be found efficiently from $\mathcal{C}$ with probability $1 - 2^{-\kappa}$.*

Proof Sketch. The proof is analogous to the proof of Lemma 2.4. It is not so hard to see that we can use sampling to find a good enough approximation for $\varphi$ in the proof of Lemma 2.4. □

Claim 2.16. *Let $\mathcal{C}$ be a collection of circuits such that* $\text{Size}(\mathcal{C}) \in \text{poly}(n)$ *and such that $\frac{1}{|\mathcal{R}_n|}\big|\{x \mid N_{\mathcal{C}}(x) \leq 0\}\big| \leq \frac{7\delta}{16}$. Then there is a circuit $C'$ of size* $\text{poly}(n)$ *for which*

$$\Pr[C'(g(x)) = f(x)] > 1 - \frac{7\delta}{16}.$$

*Furthermore, $C'$ can be found efficiently from $\mathcal{C}$.*

Proof. The majority function applied to the output of all the circuits in the collection satisfies the desired properties. □

### 2.2.5 Finishing the Proof

We can now finish the proof of Lemma 2.5.

Proof (of Lemma 2.5). To get a contradiction, let $A^{(\cdot)}$ be an oracle algorithm as in Lemma 2.5. Lemma 2.9 implies that with this, the loop of Algorithm Imp$^+$ has an efficient randomized implementation which is correct for infinitely many $n$, as long as the loop is traversed at most a polynomial number of times (since in this case Size($\mathcal{C}$) must be polynomial).

Lemma 2.14 states that for those $n$ the loop is traversed at most $4\gamma^{-2}\delta^{-3}$ times. Thus for the infinitely many $n$ for which the implementation does what the idealized algorithm does, a collection of circuits will be returned which either satisfies the condition of Claim 2.15 or 2.16, and it is easy to decide which is the case. One of those claims can then be used to to get a circuit $C'$ which satisfies for some noticeable function $\varphi$

$$\Pr[C'(g(x)) = f(x)] > 1 - \frac{\delta}{2} + \varphi.$$

The probability that anything does not run correctly can easily be bounded by $2^{-n}$ using these Lemmas. Since the circuit $C'$ can then be simulated, we get the contradiction (for infinitely many $n$).

Furthermore, since the running time of the algorithm is polynomially bounded we can count the number of steps, and after a suitable polynomial number of steps stop the algorithm. This ensures that the running time is polynomial for all $n$, while the success can still be achieved for infinitely many $n$. □

## 2.3 Computational and Information Theoretic Key Agreement

Lemma 2.5 is sufficient to show that any efficient information theoretic SKA protocol for $(\varepsilon, \delta)$-secure random variables can be used without modification to improve a given $\delta$-secure SBA protocol with correlation $\varepsilon$ to a key agreement protocol. Note that in the following theorem, $\varepsilon$ and $\delta$ can be arbitrary functions as long as they are efficiently computable.

Theorem 2.17. *Let $\delta : \mathbb{N} \to [0, 1]$, $\varepsilon : \mathbb{N} \to [0, 1]$ be computable in time* poly($n$). *If there exists a $\delta$-secure SBA protocol with correlation $\varepsilon$, and there exists an efficient information theoretic secret-key agreement protocol for $(\varepsilon, \delta)$-secure random variables, then there exists a (computational) key agreement protocol.*

Proof. We assume without loss of generality that the given SBA protocol uses $n$ bits of randomness per invocation, and $n$ bits of communication per invocation (any protocol which does not satisfy this can be easily modified by padding to satisfy this). Furthermore, define $\mathcal{R}_n \subseteq \{0, 1\}$ to be the set of randomness for which $S_A = S_B$ after the SBA protocol. We note that $|\mathcal{R}_n| \geq 2^{n-1}$. Let $g : \{0, 1\}^n \to \{0, 1\}^n$ be the communication the SBA protocol generates, and let $f : \{0, 1\}^n \to \{0, 1\}$ be the secret bit $S_A$ for this randomness.

Let $k \in \text{poly}(n)$ be the maximal number of instances the given information theoretic secret-key agreement protocol may use. Alice and Bob first use the SBA protocol $k$ times to obtain bits $X_1, \ldots, X_k$ and $Y_1, \ldots, Y_k$, while producing communication $\Gamma_1, \ldots, \Gamma_k$. They use the bits as inputs to the information theoretic secret-key protocol and run it. The outputs $X$ and $Y$ of the information theoretic protocol are then the outputs of the new protocol, and let $\Gamma$ be the communication produced by the information theoretic protocol (i.e., $\Gamma$ does not include $\Gamma_1, \ldots, \Gamma_k$).

It is obvious that the the protocol can be run in time poly($n$), and also that the probability that the output bits of Alice and Bob are equal is at least $1 - 2^{-n}$.

In order to prove the security of the protocol, we show that an algorithm $A$ which breaks the resulting protocol for infinitely many $n$ can be used to break the SBA protocol for infinitely many $n$. For that, assume that

$$\Pr[A(\Gamma_1, \ldots, \Gamma_k, \Gamma) = S_A] > \frac{1+\gamma}{2},$$

where $\gamma$ is non-negligible.

We want to use $A$ to give an algorithm which finds $f(x)$ given $g(x)$ with probability $1 - \delta/2$ for infinitely many $n$. Lemma 2.5 implies that it is sufficient to find, for any set $\mathcal{S}$ of size at least $\delta|\mathcal{R}_n|$ and some non-negligible $\gamma'$, a circuit $C$ such that $\mathrm{E}[\Pr_{x \leftarrow \mathcal{S}}[C(g(x)) = f(x)]] \geq (1 + \gamma')/2$. For this, a given oracle $\chi_S$ can be used.

Let now such a set $\mathcal{S}$ with be fixed. We will use the the hybrid argument to find the circuit. First, we describe $k+1$ different variations of protocol. For variation $j$, $0 \leq j \leq k$, we first run the SBA protocol $k$ times with uniformly chosen randomness $r_1, \ldots, r_k$ to obtain $X_1, \ldots, X_k$, $Y_1, \ldots, Y_k$ and $\Gamma_1, \ldots, \Gamma_k$. Next, for all $i \leq j$ with $X_i = Y_i$ and $r_i \in \mathcal{S}$, we replace $X_i$ and $Y_i$ with the a uniform random bit (i.e., choose $R_i$ at random and set $X_i = R_i$ and $Y_i = R_i$). Then we continue with the information theoretic protocol. Let $p_j$ be the probability that $A(\Gamma_1, \ldots, \Gamma_k, \Gamma) = S_A$, for an execution of protocol $j$. Obviously, $p_0 \geq (1 + \gamma)/2$ and $p_k \leq (1 + 2^{-n})/2$, since in protocol $k$, $(X_i, Y_i, \Gamma_i)$ are $(\varepsilon, \delta)$-secure random variables. A simple application of the hybrid argument now implies that we can obtain a circuit $C'$ such that

$$\Pr_{x \leftarrow \mathcal{S}}[C'(g(x)) = f(x)] \geq \frac{1 + \gamma/k - 2^{-n}}{2}.$$

Lemma 2.5 now proves the Theorem. □

# 3. INFORMATION-THEORETIC ASPECTS

We show that an information theoretic secret-key agreement protocol for $(\varepsilon, \delta)$-secure random variables exists if $\delta > \frac{1-\varepsilon}{1+\varepsilon}$. It is convenient to use a different parametrization in this section. Namely, we set $\vartheta := \frac{1-\varepsilon}{1+\varepsilon}$. For reference, this gives the following conversion formulas:

$$\vartheta = \frac{1-\varepsilon}{1+\varepsilon}$$
$$\varepsilon = \frac{1-\vartheta}{1+\vartheta}$$
$$\Pr[X = Y] = \frac{1}{1+\vartheta} = \frac{1+\varepsilon}{2}.$$

Note that the goal is to make $\vartheta \in [0, 1]$ as small as possible. With this parametrization, we use brackets to characterize random variables:

Definition 3.1. *Let $\mathcal{Z}$ be any set. A triple $X \times Y \times Z$ of random variables over $\{0, 1\} \times \{0, 1\} \times \mathcal{Z}$ is $[\vartheta, \delta]$-secure if it is $(\frac{1-\vartheta}{1+\vartheta}, \delta)$-secure.*

Our information theoretic protocol consists of two building blocks:

- Alice and Bob compute the XOR of multiple instances, and keep the resulting bit as new random variable.
- Alice and Bob compute the XOR of two instances, and Alice sends the resulting bit to Bob. Bob checks whether he gets the same, and tells this to Alice. They

keep the first of the two instances if they had the same bit.

We will show that after each of these operations, when used with $[\vartheta, \delta]$-secure random variables, Alice and Bob again share $[\vartheta', \delta']$-secure random variables, and give bounds on $\vartheta'$ and $\delta'$. Note that in order to obtain efficient secret-key agreement protocol, it is sufficient to construct $[2^{-n}, 1-2^{-n}]$-secure random variables for every $n$ in an efficient way.

Definition 3.2. *A $[\vartheta, \delta]$ to $[\vartheta', \delta']$* conversion protocol *with cost c is a protocol, such that, for independent $[\vartheta, \delta]$-secure random variables $(X_i, Y_i, Z_i)$, $i \in \mathbb{N}$:*

- *Alice and Bob use R random variables $(X_1, Y_1, Z_1), \ldots, (X_R, Y_R, Z_R)$ and $\mathrm{E}[R] \leq c$.*
- *The protocol can be computed in time polynomial in c.*
- *Alice and Bob output a random variable $(X', Y')$, and for the communication $\Gamma$ the triple $(X', Y', Z_1 \ldots Z_R\Gamma)$ is $[\vartheta', \delta']$-secure.*

## 3.1 The XOR of Multiple Variables

If Alice and Bob just take the XOR of $r$ instantiations this gives a $[\vartheta, \delta]$ to $[r\vartheta, r\delta(1-2r\delta)]$ conversion. In order to prove this, we need a few technical lemmas.

The following inequality is similar to Bernoulli's inequality.

Lemma 3.3. *Let $r \in \mathbb{N}$, $\vartheta > 0$. Then,*

$$\Big(\frac{1-\vartheta}{1+\vartheta}\Big)^r \geq \frac{1-r\vartheta}{1+r\vartheta}.$$

Proof. Using induction on $r$. □

This allows us to compute the probability that the XOR of multiple i.i.d. $\{0,1\}$ random variables is 1.

Lemma 3.4. *Let $R_1, \ldots, R_r$ be i.i.d. random variables over $\{0,1\}$ with $\Pr[R_i = 0] = \frac{1}{1+\vartheta}$. Then*

$$\Pr[R_1 \oplus \cdots \oplus R_r = 0] = \frac{1}{2} + \frac{1}{2}\Big(\frac{1-\vartheta}{1+\vartheta}\Big)^r \geq \frac{1}{1+r\vartheta}$$

Proof. The equality is proven by induction over $r$. The inequality follows directly from Lemma 3.3. □

Lemma 3.5. *Let $\delta$, $\vartheta$ be given. For every $r$ there exists a $[\vartheta, \delta]$ to $[r\vartheta, (1-r\vartheta)(1-(1-\delta)^r)]$ conversion with cost $r$.*

Proof. Let $(X_i, Y_i, Z_i)$ with the corresponding event $\mathcal{E}_i$, for $1 \leq i \leq r$, be $r$ instances of $[\vartheta, \delta]$-secure random variables. Alice and Bob compute the parity $X = X_1 \oplus \cdots \oplus X_r$ and $Y = Y_1 \oplus \cdots \oplus Y_r$, respectively, and we show that the tuple $(X, Y, Z_1 \ldots Z_r)$ is $[r\vartheta, (1-r\vartheta)(1-(1-\delta)^r)]$-secure.

Lemma 3.4 applied on $R_i := X_i \oplus Y_i$ implies $\Pr[X = Y] \geq \frac{1}{1+r\vartheta}$. We define the event $\mathcal{E} := (\forall i : X_i = Y_i) \wedge (\exists i : \mathcal{E}_i)$. Obviously $\mathcal{E}$ implies $X = Y$ and $I(X; Z_1 \ldots Z_r \mid \mathcal{E}) = 0$. We obtain

$$\begin{aligned}\Pr[\mathcal{E} \mid X = Y] &\geq \Pr[\mathcal{E}]\\ &= \Pr[\forall i : X_i = Y_i]\Pr[\mathcal{E} \mid \forall i : X_i = Y_i]\\ &\geq (1+\vartheta)^{-r}(1-(1-\delta)^r)\\ &\geq (1-r\vartheta)(1-(1-\delta)^r). \quad \Box\end{aligned}$$

As a corollary we obtain:

Corollary 3.6. *For every $\vartheta$, $\delta$ with $\delta > 100\vartheta$ there exists a $[\vartheta, \delta]$ to $[\frac{1}{16}, \frac{15}{16}]$ conversion with cost $\frac{5}{\delta}$.*

Proof. Since $1-(1-\delta)^r \geq 1-e^{-r\delta}$, Lemma 3.5 also gives a $[\vartheta, \delta]$ to $[r\vartheta, (1-r\vartheta)(1-e^{-r\delta})]$ conversion. Using $r = \frac{5}{\delta}$ implies that $r\vartheta < 1/20$ and $(1-r\vartheta)(1-e^{-r\delta}) > 15/16$. □

## 3.2 The XOR with Communication

Next we consider the protocol where Alice and Bob first both compute the XOR of two random variables and then use the authentic channel to communicate it. In case the XOR is the same for both, they keep the first of the two initial bits, and otherwise they repeat the protocol.

Lemma 3.7. *Let $\vartheta, \delta$ with $\delta > \vartheta$ be given. There exists a $[\vartheta, \delta]$ to $[\vartheta^2, \delta^2]$ conversion with cost $2(1+\vartheta)/(1-\vartheta)$.*

Proof. Assume that $(X_1, Y_1, Z_1)$ and $(X_2, Y_2, Z_2)$ are $[\vartheta, \delta]$-secure. Alice sends $X_1 \oplus X_2$ to Bob, who checks if this is equal to $Y_1 \oplus Y_2$. If this is the case he notifies Alice that the protocol was successful, and they output $X = X_1$ and $Y = Y_1$, respectively. Otherwise they discard the bits and start over again. Note that Eve will know at which point Alice and Bob accepted.

It is easy to see that the probability that Alice and Bob accept is $\frac{1-\vartheta}{1+\vartheta}$, and the probability that $X_1 = Y_1$ and $X_2 = Y_2$ is $(1+\vartheta)^{-2}$, which implies that the probability that $X = Y$ holds after the protocol is $\frac{1}{1+\vartheta^2}$.

We define $\mathcal{E} := \mathcal{E}_1 \wedge \mathcal{E}_2$. It is obvious that $\mathcal{E}$ implies $X = Y$ and $I(X; Z_1 Z_2 C \mid \mathcal{E}) = 0$. Since $\Pr[\mathcal{E} \mid X = Y] = \Pr[\mathcal{E} \mid (X_1 = Y_1) \wedge (X_2 = Y_2)] \geq \delta^2$ we see that the protocol gives a $[\vartheta, \delta]$ to $[\vartheta^2, \delta^2]$ conversion. The expected number of repetitions of the protocol is $\frac{1+\vartheta}{1-\vartheta}$ and thus the cost is $2(1+\vartheta)/(1-\vartheta)$. □

## 3.3 Combining the protocols

First we show that we can increase the security, once we have $[\frac{1}{16}, \frac{15}{16}]$-secure random variables.

Lemma 3.8. *For every $\frac{1}{16} \geq \vartheta > 0$, $\overline{\delta} > 0$ there exists a $[\vartheta, 1-\overline{\delta}]$ to $[2\vartheta^2, 1-2\vartheta^2-4\overline{\delta}^2]$-conversion with cost 5.*

Proof. We first use Lemma 3.7 to get $[\vartheta^2, (1-\overline{\delta})^2]$-secure random variables. Since $(1-\overline{\delta})^2 \geq 1-2\overline{\delta}$, these random variables are also $[\vartheta^2, 1-2\overline{\delta}]$-secure. We then use two of the resulting instances with Lemma 3.5 for $r = 2$, to obtain $[2\vartheta^2, (1-2\vartheta^2)(1-4\overline{\delta}^2)]$-secure random variables, which are also $[2\vartheta^2, 1-2\vartheta^2-4\overline{\delta}^2]$ secure. Finally, $\frac{1}{16} \geq \vartheta$ implies that the cost is at most 5. □

Lemma 3.9. *Let $\delta$, $\vartheta$ with $\delta > \vartheta$ be given. For every $n > 0$ there exists a $[\vartheta, \delta]$ to $[2^{-n}, 1-2^{-n}]$ conversion with cost $c_{\vartheta,\delta} \cdot n^3$, where $c_{\vartheta,\delta}$ depends only on $\vartheta$ and $\delta$.*

Proof. Since $\delta > \vartheta$, Lemma 3.7 implies that there exists a $[\vartheta, \delta]$ to $[\vartheta', \delta']$ conversion with $\delta' > 100\vartheta'$. Together with Corollary 3.6 this implies that there exists a $[\vartheta, \delta]$ to $[\frac{1}{16}, \frac{15}{16}]$ conversion with constant cost $c_{\vartheta,\delta}$.

Starting from $[\frac{1}{16}, \frac{15}{16}]$-secure variables, it is easy to verify that $s$ iterations of Lemma 3.8 yield $[2^{-3-2^s}, 1-2^{-3-2^s}]$-secure random variables at a cost $5^s$. Letting $s = \log n$ this implies the lemma. □

### 3.4 Impossibility

In this section, we show that we can generate $[\vartheta, \delta]$-secure random variables from scratch (using only an authentic channel), as long as $\vartheta \geq \delta$. Since the protocol obviously then satisfies the requirements for a $\delta$-secure SBA protocol with correlation $\varepsilon = (1-\vartheta)/(1+\vartheta)$, this implies that if one can give a reduction from such a protocol to secret-key agreement one automatically obtains a protocol for secret-key agreement.

LEMMA 3.10. *For any $1 \geq \vartheta \geq \delta \geq 0$, there exists a protocol for Alice and Bob with output $(X, Y)$ using only an authentic channel and two bits of communication $C$ such that $(X, Y, C)$ is a $[\vartheta, \delta]$-secure random variable.*

PROOF. It is sufficient to consider the case $\delta = \vartheta$. In this case, Alice chooses a bit $b_0$ such that $\Pr[b_0 = 0] = 2\delta/(1+\delta)$, a bit $b_1$ such that $\Pr[b_1 = 0] = \Pr[b_1 = 1] = \frac{1}{2}$, and sends both bits to Bob. If $b_0 = 0$, both players output a random bit, if $b_0 = 1$ they output $b_1$.

The probability that they output the same bit is $\delta/(1+\delta)+(1-\delta)/(1+\delta) = 1/(1+\delta)$. The event $\mathcal{E}$ is defined as $(b_0 = 0)\wedge(X = Y)$, and $\Pr[\mathcal{E} \mid X = Y] = (\delta/(1+\delta))(1+\delta) = \delta$, which implies that $(X, Y, C)$ is a $[\delta, \delta]$-secure random variable. □

This lemma could also be obtained by observing that for $[\vartheta, \delta]$-secure random variables with $\vartheta \geq \delta$ the intrinsic information is zero. Techniques implicit in [16] show that the information of formation of this distribution must then be zero as well.

In total, we obtain the main theorem of this section:

THEOREM 3.11. *For constants $0 \leq \delta, \varepsilon \leq 1$, there exists an information theoretic secret-key agreement protocol for $(\varepsilon, \delta)$-secure random variables if and only if $\delta > \frac{1-\varepsilon}{1+\varepsilon}$.*

PROOF. Let the given security parameter be $n$. First we note that $(\varepsilon, \delta)$-secure random variables with $\delta > \frac{1-\varepsilon}{1+\varepsilon}$ are $[\vartheta, \delta]$ secure random variables with $\delta > \vartheta$. Lemma 3.9 implies that a secret-key agreement protocol with expected polynomial cost (in $n$) exists. Such a protocol is easily modified to one where the worst case cost is polynomial, and such that it only fails with probability $2 \cdot 2^{-n}$.

On the other hand, if $\vartheta \geq \delta$, Lemma 3.10 implies (cf. [13]) that no such protocol is possible. □

Theorem 3.11 can be combined with Theorem 2.17 to get the main result:

THEOREM 3.12. *For constants $0 \leq \delta, \varepsilon \leq 1$, there exists a relativizing reduction from key agreement to a $\delta$-secure SBA protocol with correlation $\varepsilon$ if and only if $\delta > \frac{1-\varepsilon}{1+\varepsilon}$.*

PROOF. If $\delta > \frac{1-\varepsilon}{1+\varepsilon}$ the claim follows directly from Theorems 2.17 and 3.11. If $\delta \leq \frac{1-\varepsilon}{1+\varepsilon}$ such a reduction would imply that secret-key agreement exists, since Lemma 3.10 implies the existence of a $\delta$-secure SBA protocol with correlation $\frac{1-\delta}{1+\delta}$. □

## Acknowledgments

Foremost I would like to thank Renato Renner for endless discussions, in particular about the contents of Section 3. Also I would like to thank Dominic Jost, Ueli Maurer, Krzysztof Pietrzak and Jürg Wullschleger for helpful comments and discussions.

The author was supported by the Swiss National Science Foundation, project no. 200020-103847/1.

## 4. REFERENCES

[1] R. Ahlswede and I. Csiszár. Common randomness in information theory and cryptography—Part I: secret sharing. *IEEE Trans.on Inf.Th.*,39(4):1121–1132.1993.

[2] T. Cover and J. Thomas. *Elements of Information Theory.* John Wiley & Sons, Inc., first edition, 1991. ISBN 0-471-06259-6.

[3] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Trans. on Inf. Theory*, IT-22(6):644–654, 1976.

[4] C. Dwork, M. Naor, and O. Reingold. Immunizing encryption schemes from decryption errors. In C. Cachin and J. Camenisch, ed., *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 342–360, 2004.

[5] O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *JACM*, 33(4):792–807, 1986.

[6] O. Goldreich, N. Nisan, and A. Wigderson. On Yao's XOR-lemma. ECCC Report TR95-050, 1995.

[7] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *Siam J. on Comp.*, 28(4):1364–1396, 1999.

[8] R. Impagliazzo. Hard-core distributions for somewhat hard problems. In *36th FOCS*, pages 538–545, 1995.

[9] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography. In *30th FOCS*, pages 230–235, 1989.

[10] R. Impagliazzo and S. Rudich. Limits on the provable consequences of one-way permutations. In *21st STOC*, pages 44–61, 1989.

[11] A. R. Klivans and R. A. Servedio. Boosting and hard-core sets. In *40th FOCS*, pages 624–633, 1999.

[12] M. Luby and C. Rackoff. How to construct pseudorandom permutations from pseudorandom functions. *Siam J. on Comp.*, 17(2):373–386, 1988.

[13] U. Maurer. Secret key agreement by public discussion. *IEEE Trans. on Inf. Theory*, 39(3):733–742, 1993.

[14] U. Maurer and S. Wolf. Unconditionally secure key agreement and the intrinsic conditional information. *IEEE Trans. on Inf. Theory*, 45(2):499–514, 1999.

[15] R. J. McEliece. A public key cryptosystem based on algebraic coding theory. DSN Progress Report 42-44:114–116, Jet Propulsion Lab, NASA, 1978.

[16] R. Renner and S. Wolf. New bounds in secret-key agreement: The gap between formation and secrecy extraction. In E. Biham, ed., *EUROCRYPT 2003*, volume 2656 of *LNCS*, pages 562–577, 2003.

[17] R. L. Rivest, A. Shamir, and L. M. Adelman. A method for obtaining digital signatures and public-key cryptosystems. *Comm. ACM*, 21(2):120–126, 1978.

[18] S. Rudich. The use of interaction in public cryptosystems. In J. Feigenbaum, ed., *CRYPTO '91*, volume 576 of *LNCS*, pages 242–251, 1991.

[19] R. E. Schapire. The strength of weak learnability. *Machine Learning*, 5(2):197–227, 1990.

[20] L. Trevisan. List-decoding using the XOR lemma. In *44th FOCS*, pages 126–135, 2003.

[21] J. von Neumann. Zur Theorie der Gesellschaftsspiele. *Mathematische Annalen*, 100:295–320, 1928.

[22] I. Wegener. *The Complexity of Boolean Functions.* John Wiley & Sons Ltd, and B. G. Teubner, Stuttgart, 1987. ISBN 0-471-91555-6, available on ECCC.