Low Communication Complexity Protocols, Collision Resistant Hash Functions and Secret Key-Agreement Protocols*

Shahar P. Cohen† Moni Naor‡

Abstract

We study communication complexity in computational settings where bad inputs may exist, but they should be hard to find for any computationally bounded adversary.

We define a model where there is a source of public randomness but the inputs are chosen by a computationally bounded adversarial participant *after seeing the public randomness*. We show that breaking the known communication lower bounds of the private coins model in this setting is closely connected to known cryptographic assumptions. We consider the simultaneous messages model and the interactive communication model and show that for any non trivial predicate (with no redundant rows, such as equality):

1. Breaking the $\Omega(\sqrt{n})$ bound in the simultaneous message case or the $\Omega(\log n)$ bound in the interactive communication case, implies the existence of distributional collision-resistant hash functions (dCRH). This is shown using techniques from Babai and Kimmel [BK97]. Note that with a CRH the lower bounds can be broken.

2. There are no protocols of constant communication in this preset randomness settings (unlike the plain public randomness model).

The other model we study is that of a stateful "free talk", where participants can communicate freely *before* the inputs are chosen and may maintain a state, and the communication complexity is measured only afterwards. We show that efficient protocols for equality in this model imply secret key-agreement protocols in a constructive manner. On the other hand, secret key-agreement protocols imply optimal (in terms of error) protocols for equality.

*Research supported in part by grants from the Israel Science Foundation (no.2686/20) and by the Simons Foundation Collaboration on the Theory of Algorithmic Fairness.

†Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, Israel. Email: shahar.cohen@weizmann.ac.il.

‡Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, Israel. Incumbent of the Judith Kleeman Professorial Chair. Email: moni.naor@weizmann.ac.il.

Contents

1 Introduction

What does a lower bound mean if it is not feasible to find the bad inputs? In other words, can we bypass it, if we assume that the choice of inputs is done by a process that is computationally limited? In this work we study this issue in the setting of communication complexity.

The study of communication complexity deals with proving bounds on the amount of communication that is required to perform certain tasks when the input is separated: two parties, Alice and Bob, have as inputs $x \in X$ and $y \in Y$ respectively; how many bits do they have to send each other (as a function of $|X|$ and $|Y|$) for computing the value of a function $f(x, y)$? An answer for such a question depends, of course, on the exact model details: Do Alice and Bob have any limitation in the communication? Do they communicate directly or through a third party? What predicates f are they trying to compute? See Kushilevitz and Nisan [KN96] and Rao and Yehudayoff [RY20] for background on communication complexity.

There are several models of communication that differ mainly on two properties: whether the strategy of the participants can be probabilistic and the exact communication settings (network layout). The participants of those models do not have a bound on their running time, however, they are required to be correct[1] for every input in the space.

When the participants are allowed to be probabilistic there is an important distinction: whether they share common random bits (public coins) or not (private coins). It is important to note that the random bits (in both options) and the problem's inputs (x and y) are *independent*. This can be seen as uniform random bits that are chosen *after* the (worst-case) input was chosen.

By definition, the private coins model is no stronger than the public coins model and indeed some tasks can be done in the latter but cannot be done in the former with the same communication complexity (see later). On the other hand, the private coins model may be considered more *realistic*, where there is no *independent* public random string.

However, both the public and private coins models are known to be 'better' than the deterministic model in the sense that they have more efficient protocols in terms of the communication complexity: for instance, as proved by Yao, the deterministic communication complexity of many predicates is $\Omega(n)$ (Alice and Bob can do nothing better than just sending their full inputs), while in the probabilistic world there is quite a lot to be done. Equality is a prominent example with $O(\log n)$ and $O(1)$ algorithms respectively.

We examine another relaxation that can help us: limiting all parties, including the one who selects the inputs to a computationally bounded world. We will not require that Alice and Bob be correct for every input in the space, but only to inputs that are chosen by a computationally bounded adversary. Note that the new definition is by nature relevant only to probabilistic algorithms.

Considering a polynomially bounded adversary raises the question of whether there are benefits from different computational hardness assumptions: can we reduce our communication complexity by assuming that a certain task cannot be performed efficiently? That is, given that Alice and Bob in our new definition do not have to be correct for every input in the input space, a computational hardness assumption can be used for proving that no

[1] For the probabilistic version they are required to succeed with constant high probability.

efficient adversary can find bad inputs with non-negligible probability.

Back to the relationship between the public and private coins models: we propose a new model that is, in general, more powerful than the private coins but still realistic. Also, in contrast to the above mentioned models, our model is *computational* – the participants' running time is bounded by some poly(λ) where λ is the security parameter. In our model, there is a public random string but there is an additional adversarial participant that *chooses the inputs depending on the public random string*, it can be seen as a public random string that is 'fixed' in advance and therefore we called it *preset public coins* model. See Definition 2.9 for formal specification.

The two communication patterns we consider are:

Simultaneous Messages Model. In the simultaneous messages (SM) model Alice and Bob are given x and y respectively and should compute some function f but *without communicating with each other*. Instead, each one sends a message to a third party (a referee) who calculates $f(x, y)$ given the messages from Alice and Bob.

Interactive Communication Model. Alice and Bob get their inputs and can communicate with each other without any limitations on the number of rounds.

In both settings, the communication complexity measure is the total length of the messages sent by Alice and Bob.

Stateful preprocessing communication. The second type of model we consider is where the communication complexity matters *only at some critical period of time.* The two parties can talk freely beforehand. At some point the action starts, they receive their inputs and need to decide with little communication the result. In the SM model we also consider a variation that differs by two properties:

Free talk. A protocol with *free talk* is one where Alice and Bob communicate also before getting their inputs. The messages during the free talk phase (before the inputs are chosen) do not count in the communication complexity of the protocol. However the adversary sees the whole communication and can use it while he chooses the inputs. Alice and Bob maintain (secret) states afterwards.

Rushing adversary. In our model the inputs are chosen by a computationally bounded adversary depending on the public random string. A rushing adversary can choose Bob's input at the 'last moment': He first chooses the input of Alice depending on the public random string and *afterwards* chooses the input of Bob depending on both the public random string *and Alice's message.*

1.1 Cryptographic Primitives

We discuss a computationally bounded world. We assume that all parties have limited resources (especially at runtime). A way to express those limits is by cryptographic primitives (see the next examples).

Necessity of Primitives. One of the aims of research in foundations of cryptography is to find out which cryptographic primitives are essential and sufficient for which tasks. Similarly, it is valuable to know whether certain primitives on their own *cannot* help us achieve a certain goal.

In this paper we prove several implications of the form that the existence of communication protocols with certain properties entails the existence of certain primitives. In other words, in order to desgin succinct protocols in those models we must be using somewhere in the protocol primitives of a certain kind.

1.2 Cryptographic Hash Functions

A hash function is one that maps values from a large domain to a smaller range. One of the most basic cryptographic objects is a hash function with some hardness property. For instance, a family of hash functions $\mathcal{H}$, is collision resistant if for a random $h \in_R H$ it is hard to find two inputs $x \neq y$ that collide ($h(x) = h(y)$).

For such a function, for any two inputs that were chosen by a computationally bounded adversary, we know that w.h.p., $h(x) = h(y) \implies x = y$. This means that the function *preserves* some relation between its inputs: The equality predicate is (w.h.p.) preserved also after the values were compressed by h. Moreover, since the function is collision resistant, that property holds for any (x, y) chosen by a computationally bounded adversary knowing h.

This notion can be generalized in several directions:

1. More relaxed hardness requirements can be defined. The weaker the definition the more hope we have to construct it from minimal assumptions.

2. We can extend the definitions to include random algorithms: functions that get also random bits and output correct values w.h.p.[2]

3. We can extend the definitions to hash functions that preserve more properties and not just the equality predicate.

We discuss the last two points in the section below.

1.3 Adversarially Robust Property-Preserving Hash Functions

Consider a predicate $P\colon U \times U \to \{0,1\}$ for a universe $U = \{0,1\}^n$. Let $x, y \in U$ and we want to compute $P(x, y)$, but we cannot have both x, y on the same machine (say, for some storage reasons). A natural approach for this issue is using *sketching*: By using sketches we get shorter strings and it is easier to get both (sketched) values on the same machine. Of course, computing P on sketched values may be impossible in terms of information, so we relax the correctness requirement: the process may fail (compute a wrong value) with at most a negligible probability[3].

[2] The probability is over the choices of the random bits.

[3] The probabilities are over the sampling of a hash function among the functions family.

Hash functions as above, that allow us to compute a predicate given the hashed values, are called *property-preserving* hash functions (hereafter PPH). We examine PPH in an *adversarial* environment, that is, the predicate should be computed correctly w.h.p. also for values chosen by an adversary. Such hash functions are called *adversarially robust* PPH. The more access to the hash function given to the adversary the more robust the PPH is.

The study of adversarially robust property-preserving hash functions was initiated by Boyle et al. [BLV18]. It can be seen as a special case of the model introduced by Mironov et al. [MNS11] who initiated the study of the *adversarial sketch model* (here the participants also get the input *online*). That notion is similar to the SM model except for the differences:

1. The allowed error probability in communication complexity is a (small) constant instead of negligible in PPHs.
2. The parties in the SM model are allowed to be randomized.
3. The PPHs model is computational.

Our model bridges some of the gaps and we will show the connection between the models. Note that the preset public coins SM model is a generalization of the PPHs model in the sense that the participants are allowed to be randomized. In this regards it is closer to the model of Mironov et al. [MNS11].

1.4 Secret Key Agreement

A secret key agreement (SKA) is a protocol where two parties with no prior common information agree on a secret key. The key has to be secret in the sense that no probabilistic polynomial time adversary given the full transcript of the communication between Alice and Bob can compute it with non-negligible probability (more accurately, distinguish it from a random string). That notion is defined formally in Definition 2.15.

We will show that certain low communication protocols imply the existence of SKA by showing a construction of SKA from those protocols.

1.5 Our Results

We consider preset public coins communication complexity models and prove that the lower bounds proved for the private coins model cannot be broken in our computational model without assuming the existence of distributional CRHs (dCRH is a hash function where uniformly random collisions cannot be found by a bounded adversary w.h.p., see Definition 2.13) It is known that dCRHs exist only if one-way functions exist and there is an oracle separation between them (i.e. there are no black-box constructions of dCRHs from one-way functions).

A non-trivial predicate is one with no redundant row a columns (see Definition 2.4)

Theorem (informal, see Theorems 3.2 and 3.14)**.** *In the preset public coins Simultaneous Message model: for any non-trivial predicate, protocols with communication complexity $o(\sqrt{n})$ imply the existence of dCRHs (in the sense that a dCRH can be constructed from the protocol).*

In the interactive model: The same is true for $c(n) = o(\log n)$.

Table 1: Summary of Implications Results

		Information-Theoretic Lower Bound	Computationally Bounded World: Breaking The Bound	
			Possible Using	Implies
Stateless	SM	$\Omega(\sqrt{n})$	CRH	dCRH
	Interactive	$\Omega(\log n)$	CRH	dCRH
Stateful (Rushing $\mathcal{A}$dv)	SM	$\Omega(n)$	SKA	For Equality: SKA*

* Holds only for near optimal protocols.

Consider the free talk model, where two parties communicate and may have a secret state as a result, *before* the inputs are chosen based on an eavesdropper adversary who has access to the communication but not to the secret states. If secret key agreement protocols exist, then we can get the power of the public coins model: we can construct a protocol for the equality predicate with error probability bounded by 2^{-c} where c is the communication complexity.

In the other direction, nearly optimal protocols imply a secret key agreement:

Theorem (informal, see Theorem 5.2)**.** *In the stateful free talk model, the existence of a protocol of complexity $c(n)$ for equality with failure probability bounded by $\varepsilon \leq 2^{-0.7c}$ against a rushing adversary implies the existence of a secret key agreement protocol.*

Our implications results are summarized in Table 1.

On the other hand, regardless of assumptions, constant communication protocols cannot exist in our model. This is in contrast to the public coins model, where there are protocols of $O(1)$ communication even in the SM model (e.g. for the equality predicate).

Theorem (informal, see Theorem 4.1)**.** *In the interactive communication model, protocols for any non trivial predicate, of communication complexity $O(\log \log n)$ bits are not secure against an adversary with running time* $\mathrm{poly}(n)$*.*

1.6 Related Works

Grossman et al. [GHY20] studied a similar notion in the context of error correcting codes. They studied codes that deal with errors done by a polynomial time process. Grossman et al. relied on some cryptography assumptions to construct a code better than codes for worst-case errors. Next, we discuss related works in the context of communication complexity.

1.6.1 Communication Complexity

The study of communication complexity was initiated by Yao [Yao79] who introduced the SM private coins model and asked what is the complexity of the equality predicate in this model. The problem was solved by Newman and Szegedy [NS96] who provided the $\Omega(\sqrt{n})$ tight lower bound. It was also solved, using different and simpler techniques, by Babai and Kimmel [BK97][4] using a combinatorial proof, and by Bottesch et al. [BGK15] using information theory[5].

Babai and Kimmel's result is more general and they actually proved the lower bound not only to the equality predicate but to any non-redundant predicate (see Definition 2.4). Moreover, their technique proved to be useful in more models: Ben-Sasson and Maor [BM15] applied this technique also for the interactive model and proved that for any non redundant function, any private coins protocol requires communication complexity of at least $\Omega(\log n)$ (see proof for the equality predicate in Kushilevitz and Nisan [KN96]).

Although the above mentioned results are in the information-theoretic world (can be seen as an unbounded adversary), Naor and Rothblum [NR09] introduced and studied a computational model in order to study online memory checking algorithms: The consecutive messages model where the public coins are chosen *after* the adversary chooses x (the input for Alice). They adapted this technique and showed that breaking the mentioned information-theoretic $\Omega(\sqrt{n})$ lower bound in their computational model is possible if and only if one-way functions exist. At first glance one can think that their model is very close to our preset public coins SM model. However, important details differ: For instance, the fact that x (Alice's input) does not depend on the public random string.

Harsha et al. [HIKNV04] studied tradeoffs between communication complexity and time complexity and described Boolean functions with strong communication-runtime tradeoff.

1.6.2 Public Coins vs. Private Coins

In certain ways our model lies between the public and private coins ones. Therefore, it is worth pointing out the possible gap between them. For the interactive communication settings, Newman [New91] proved that the gap can be at most $O(\log n)$. It is tight, since the equality predicate can be computed by protocols of $O(1)$ communication in the public coins model but requires $\Theta(\log n)$ bits in the private coins model.

In the SM model, as mentioned, the gap may be much larger: the equality predicate can be computed using $O(1)$ bits in the public coins model, but in the private coins model $\Omega(\sqrt{n})$ bits are required.

1.7 Technical Overview

Babai and Kimmel's Characterizing Multiset. We will use the technique of Babai and Kimmel for proving connections between the communication complexity and cryptographic primitives in both models (SM and interactive). They proved that in the SM model, Alice's behavior can be characterized by a relatively small multiset of messages. Ben-Sasson and Maor expanded it for the interactive model and proved that Alice's behavior can be characterized by a multiset of *deterministic strategies*.

We use those observations and show that the adversary can use the characterizing multisets to find bad inputs for Alice and Bob. That is, we construct a function that for any x (Alice's input) generates a characterizing multiset of the behavior of Alice for this x. We

[4] See in [BK97] also the similar proof of Bourgain and Wigderson.

[5] Bottesch et al. actually discuss quantum variants of the SM model and give the simpler proof for our classical case as a warm-up.

claim that an adversary who can break the security of this function, can find bad inputs for the protocol. On the other direction, if such a protocol exists it implies the existence of a certain cryptographic primitive.

2 Preliminaries

2.1 Probability

To measure distance between two distributions we use the total variation distance:

Definition 2.1 (Statistical Distance)**.** Let D_1 and D_2 be two distributions and $D(E)$ is the probability of event E under the distribution D.

$$\Delta(D_1, D_2) = \max_{\text{Event } E} |D_1(E) - D_2(E)|$$

We use in proofs the following lemma:

Lemma 2.2. *Let $X_1, X_2, \ldots, X_t$ be mutually independent random variables where $\mathbf{E}[X_i] = 0$ and $|X_i| \leq 1$. Let $S = \frac{1}{t}\sum_{i=1}^{t} X_i$ then*

$$\mathbf{Pr}[S > \delta] < e^{-\delta^2 t/2}$$

which is a direct consequence of the Chernoff bound:

Theorem 2.3 (Chernoff Bound [AS08, Theorem A.1.16])**.** *Let $X_1, X_2, \ldots, X_t$ be mutually independent random variables where $\mathbf{E}[X_i] = 0$ and $|X_i| \leq 1$. Let $S = \sum_{i=1}^{t} X_i$ then*

$$\mathbf{Pr}[S > a] < e^{-a^2/2t}$$

2.2 Model Definition

Let f be a predicate that Alice and bob would like to compute. For a predicate f to be interesting we may assume that the f has no redundancy:

Definition 2.4 (Non-Redundant Predicate)**.** Predicate $f\colon X \times Y \to \{0, 1\}$ is non-redundant if there are no two identical rows or two identical columns in the truth matrix. In other words, $\forall x_1 \neq x_2 : \exists y$ s.t. $f(x_1, y) \neq f(x_2, y)$ and for $\forall y_1 \neq y_2$ as well.

Also, we discuss only predicates where their non-redundancy can be 'proven' or found efficiently:

Definition 2.5 (Efficiently Separable Predicate)**.** Let $f\colon X \times Y \to \{0, 1\}$ be a non-redundant predicate, then f is efficiently separable if there exists PPTM $\mathcal{M}$ that finds the element promised by Definition 2.4. That is $\forall x_1 \neq x_2 \in X$:

$$\mathop{\mathbf{Pr}}_{y \leftarrow \mathcal{M}(x_1, x_2)}[f(x_1, y) \neq f(x_2, y)] = 1 - \mathrm{negl}(n)$$

and similarly for $\forall y_1 \neq y_2 \in Y$ as well.

The only specific predicate we discuss is the equality predicate, $EQ(x, y) = \mathbb{1}_{\{x=y\}}$. For the equality predicate it is easy to see that both Definitions 2.4 and 2.5 hold.

Now, we define the communication layouts:

Definition 2.6 (Interactive Communication Model)**.** Alice and Bob are given x and y respectively and should compute some function f. They may send each other messages without any limit (but the total number of bits sent is the complexity).

Definition 2.7 (Simultaneous Messages (SM) Model)**.** In the simultaneous messages model, Alice and Bob are given x and y respectively and should compute some function f *without communicating with each other*. Instead, each one sends a message to a third party (a referee) who calculates $f(x, y)$ given the messages from Alice and Bob.

Following Babai and Kimmel we assume without loss of generality that the referee is deterministic.

Fact 2.8. *In the SM model there exist protocols for the equality predicate of complexity $O(\sqrt{n})$. The protocols found independently by Ambainis, Babai and Kimmel, Naor and Newman; see [BK97] for references.*

Fact 2.8 is an example of the possible gap between probabilistic and deterministic protocols in the SM model because the equality predicate is non-redundant and because of the following well known fact:

Fact. *The deterministic communication complexity of any non-redundant predicate is $\Omega(n)$.*

Now, we are ready to define our model formally, in the above described communication layouts. Recall that our model is *computational*. That is, the participants' running time is bounded by some $\mathrm{poly}(\lambda)$ for some security parameter $\lambda = \mathrm{poly}(n)$, it's important especially for the adversarial participant. That is, any PPTM run time is bounded by $\mathrm{poly}(\lambda)$.

Definition 2.9 (Preset Public Coins)**.** The preset public coins variation is defined by the following game: let Alice, Bob and the Adversary be PPTMs with running time $\mathrm{poly}(\lambda)$.

1. Public uniform random string r_{pub} is sampled[6].
2. The adversary sees r_{pub} and chooses $(x, y) \in X \times Y$.
3. Alice and Bob get (x, r_{pub}) and (y, r_{pub}) respectively.
4. Alice and Bob send message(s) (optionally using private coins) in order to compute some target function.
5. Optionally: More steps that depend on the communication settings. For instance, in the SM model the referee steps in here.

[6] Can be generalized to a sample from any known efficient distribution.

The game should succeed with probability at least $1 - \varepsilon$ for every PPTM with running time poly(λ) adversary $\mathcal{A}$dv:

$$\Pr_{\substack{r_{\text{pub}} \\ (x,y) \leftarrow \mathcal{A}\text{dv}(r_{\text{pub}}) \\ \text{Alice and Bob private coins}}} [\text{Protocol Fails}] \leq \varepsilon$$

We follow the convention in communication complexity that the error probability is required to be $\varepsilon \leq 1/3$.

Remarks. Alice's and Bob's algorithms are public and since they don't have any secret state or secret input they can be simulated (also by the adversarial participant). It is a core fact when we are using the technique of Babai and Kimmel in computational settings in the proofs of Theorems 3.2 and 3.14 and Theorem 4.1.

2.2.1 Free Talk Model

We consider a variation to the SM model where Alice and Bob are allowed to communicate freely in a preprocessing phase, before the inputs are chosen:

Free talk. Free talk is a 'free' communication that Alice and Bob can have before the inputs are chosen by the adversary. Alice and Bob can generate states (possibly secret) in the free talk phase. Those states can be used afterward to reduce the communication complexity.

However, the adversary is also stronger, in two ways:

Free Talk Eavesdropping. The transcript of the free talk phase is known to the adversary and it may choose the inputs depending also on it.

Rushing. Rushing adversary decides Bob's input at the 'last moment': Rushing adversary chooses the input of Bob *after* Alice produces its message. That is, first Alice's input is chosen and Alice sends its message, and afterwards, Bob's input is chosen depending on Alice's message and Bob sends its message.

Definition 2.10 (SM Preset Public Coins With Stateful Free Talk and Rushing Adversary)**.** Let Alice, Bob and the Adversary be PPTMs with running time poly(λ). Consider the following game:

1. Alice and Bob toss coins and communicate in order to generate their (possibly secret) states τ_A and τ_B respectively.
2. The adversary sees their full *communication* (but not their internal states τ_A and τ_B) and sets Alice's input $x \in X$.
3. Alice (that has τ_A as her internal state) gets x and sends a message m_A to the referee.
4. The adversary sees m_A and chooses Bob's input $y \in Y$, optionally depending on m_A and the free talk's transcript.

5. Bob (that has τ_B as his internal state) gets y and sends a message m_B to the referee.
6. The referee, as a function of m_A and m_B, computes the target predicate.

Alice, Bob, and the referee should compute the target predicate successfully with probability of at least $2/3$.

To simplify the analysis we assume in this model, without loss of generality, that Alice and Bob are probabilistic only in the first step. However, they can toss coins in the first step and save them in their private states for later use.

2.3 Notation

Messages Space. Denote by M_A and M_B Alice's and Bob's messages spaces. In the interactive model we consider Alice's and Bob's deterministic strategies, every strategy is represented by a rooted binary tree of depth c (the total communication): The protocol begins in the root, each vertex is owned by one party who chooses one of the children and informs the other party by sending a bit. Finally, the leaves represent the protocol's result. We denote the set of deterministic strategies of Alice and Bob by S_A and S_B respectively.

Private random string. Denote by $r_A \in R_A$ the private random string of Alice.

Public random string. Denote by $r_{\text{pub}} \in R_{\text{pub}}$ the public random string in the protocol (it is given also to the adversary).

Secret State. When Alice and Bob have secret states we denote them by τ_A and τ_B respectively.

Participant. In the SM model, for a public random string r_{pub} denote the strategy of Alice by $A_{r_{\text{pub}}}: X \times R_A \to M_A$ and Bob by $B_{r_{\text{pub}}}: X \times R_B \to M_B$. When the public random string r_{pub} is clear from the context we may omit the subscript. (When Alice and Bob have a secret state we denote Alice and Bob as a function that gets a secret state τ instead of private random string).

Referee. In the SM model denote the referee by a function $\rho_{r_{\text{pub}}}: M_A \times M_B \to \{0,1\}$ for a public random string r_{pub} or ρ when r_{pub} is clear from the context.

Communication Complexity. Denote the length of the total communication by $c = c(n, \lambda)$.

Protocol. We denote the protocol by π, and $\pi(x, y)$ denotes running the protocol on inputs x and y.

2.4 Collision Resistant Hash Functions

A collision resistant hash function (CRH) is a function that any efficient algorithm has at most a negligible probability of a collision:

Definition 2.11 (CRH)**.** Let a functions family $\mathcal{H}$ be a family of functions that (1) compress (2) are computable in polynomial time. $\mathcal{H}$ is a family of CRHs if for every polynomial $p(\cdot)$ for every PPTM $\mathcal{A}\mathrm{dv}$ and large enough λ,

$$\Pr_{\substack{h \in \mathcal{H} \\ (x,y) \leftarrow \mathcal{A}\mathrm{dv}(h)}} [x \neq y \wedge h(x) = h(y)] < \frac{1}{p(\lambda)}$$

Note that, the output of the function cannot be too small with respect to the security parameter. Otherwise, collisions can be found easily by trying sufficient inputs.

Simon [Sim98] showed that a CRH cannot be built from black-box one-way functions. Since one-way functions are existential equivalent to a lot of basic cryptographic primitives, we know that also they cannot be black-box used to construct CRHs. For an example, see Wee [Wee07] who ruled out constructions for statistically hiding commitments with low round complexity that are based only on black-box one-way functions.

2.4.1 Distributional Collision Resistant Hash Functions

Distributional collision resistant hash function (dCRH) is a function that it is hard for any adversary to generate collisions that are close to random collisions. Hence, we first have to define an ideal collision finder:

Definition 2.12 (Ideal Collision Finder $\mathcal{COL}$)**.** The random function $\mathcal{COL}$ gets a description of a hash function h and outputs (x, x') s.t. x is uniformly random and x' is uniformly random from $h^{-1}(x)$. Note that:

1. The marginal distribution of x and x' is the same: x and x' are uniformly random (but not independent).
2. It is possible that $x = x'$.

That notion of distributional collision resistance hash functions is due to Dubrov and Ishai [DI06]. However, Bitansky et al. [BHKY19] deviated from this definition and used a stronger definition[7]. Since our results hold also for the stronger definition we will use it:

Definition 2.13 (dCRH)**.** Let a functions family $\mathcal{H}$ be a family of functions that (1) compress (2) are computable in polynomial time. $\mathcal{H}$ is a family of distributional CRHs if there exists some polynomial $p(\cdot)$ s.t. for every PPTM $\mathcal{A}\mathrm{dv}$, and large enough λ,

$$\Delta(\mathcal{COL}(h), \mathcal{A}\mathrm{dv}(h)) \geq \frac{1}{p(\lambda)}$$

where $h \leftarrow \mathcal{H}$.

[7] By switching the order of quantifiers, they require one polynomial for any adversary and not that for any adversary there exists a polynomial. See the comparison in [BHKY19].

This definition is a generalization of distributional one-way functions[8] and hence implies it. Furthermore, Bitansky et al. showed that dCRHs can be used for applications that one-way functions aren't known to achieve (and are black-box separated) [BHKY19].

Although dCRH is much weaker than CRH, as noted by Dubrov and Ishai [DI06], the black-box separation result of Simon [Sim98] applied also for dCRH: Its collision finder is the same as $\mathcal{COL}$ in our definition (Definition 2.12). Simon proved that relative to $\mathcal{COL}$ one-way functions exist (although (d)CRHs do not).

2.5 Secret Key Agreement

In a secret key agreement protocol two participants who do not have a common secret, but each one has its own source of randomness, both output a value (the secret). The participants' output has to satisfy two properties: it should be the same value for the two participants (agreement), and it has to be unknown to any efficient observer (secrecy). We follow the definition of Holenstein [Hol05]:

Definition 2.14 ((α, β)-Secret Bit Agreement (SBA))**.** An efficient two party protocol without input (aside from the security parameter λ), with one bit output for each participant b and b' respectively where $b, b' \in \{0, 1\}$ is an (α, β)-secret bit agreement if

$$\mathbf{Pr}\left[b = b'\right] \geq \frac{1+\alpha}{2}$$

and for every PPTM $\mathcal{A}$dv with running time bounded by poly(λ)

$$\mathbf{Pr}\left[\mathcal{A}\mathrm{dv}(\tau) = b \mid b = b'\right] \leq 1 - \frac{\beta}{2}$$

where τ is the complete transcript of the protocol.

The previous definition is a weaker notion of the usually desirable stronger notion:

Definition 2.15 (Secret Key Agreement)**.** (α, β)-secret bit agreement is a secret key agreement protocol if $\alpha = 1 - \mathrm{negl}(\lambda)$ and $\beta = 1 - \mathrm{negl}(\lambda)$.

Holenstein [Hol06] proved when an (α, β)-secret bit agreement can be amplified efficiently to a secret key agreement:

Theorem 2.16 ([Hol06, Corollary 7.5])**.** *Let efficiently computable functions $\alpha(\lambda), \beta(\lambda)$, be given such that*

$$\frac{1-\alpha}{1+\alpha} < \beta$$

Let $\varphi = \max(2, \frac{8}{\log\left(\frac{\beta(1+\alpha)}{1-\alpha}\right)})$ and $\gamma = \frac{1}{\log(1+((1-\alpha)/(1+\alpha))^{\varphi})}$, and assume that $\frac{\varphi \cdot 2^{4\gamma}}{\alpha} \in \mathrm{poly}(\lambda)$. If there exists an (α, β)-secret bit agreement protocol for all but finitely many k, then there exists a computationally secure key agreement.

[8]Functions where it is hard to sample uniformly from $h^{-1}(h(x))$ for random x. Such functions are known to exist if and only if one-way functions exist [IL89]. (in contrast to dCRH).

Note that a secret key agreement is unlikely to be based (only) on one-way permutations and collision resistant hash functions in a black-box manner: It is known that any secret key agreement protocol in the random oracle model[9] can be broken using an $O(n^2)$ queries attack [IR89; BM09] and this is tight.

3 Collision Resistance and the Preset Public Coins model

3.1 CRHs imply succinct protocols

We start by noting that the lower bounds of $\Omega(\sqrt{\lambda})$ in the SM model (Theorem 3.2) and of $\Omega(\log\lambda)$ in the interactive model (Theorem 3.14) shown in Section 3.2 are tight, if one is using CRHs. See Appendix B.2 for details.

Theorem 3.1. *If CRHs exist, then given a family of CRHs $\{h\colon \{0,1\}^n \to \{0,1\}^\lambda\}$,*

In the preset public coins SM model: *There exist protocols of complexity $O(\sqrt{\lambda})$ for the Equality predicate.*

In the preset public coins interactive model: *There exist protocols of complexity $O(\log\lambda)$ for the Equality predicate.*

3.2 Succinct Protocols Imply dCRHs

Theorem 3.2. *Let $\omega(\sqrt{\lambda}) \le c(n) \le o(\sqrt{n})$. Given a protocol for an efficiently separable predicate (Definition 2.5) of complexity $c(n)$ in the preset public coins SM model, we can construct a distributional CRH.*

Proof. Our intuition is that after fixing the public random string r_{pub}, the model is similar to the private coins SM model where the adversary is faced with a problem defined by the random string. We therefor appearl to Babai and Kimmel's definitions and techniques. Furthermore, in Lemma 3.5 we will also repeat the proof of [BK97, Lemma 2.3] with a different constant and make it constructive.

For each multiset of Alice's messages and one message from Bob we consider the probability of acceptance by the referee:

Definition 3.3 (Referee's Expected Value for a Multiset)**.** For any r_{pub}, for a multiset T of members from M_A and $m_B \in M_B$, let

$$Q(T, m_B) = \mathop{\mathbf{E}}_{i\in[t]}[\rho_{r_{\text{pub}}}(T[i], m_B)] = \frac{1}{t}\sum_{i\in[t]} \rho_{r_{\text{pub}}}(T[i], m_B)$$

where $t = |T|$.

[9] CRHs exist in this model.

Now, we show that for every input of Alice $x \in X$, there exists a multiset *characterizing* the behavior of Alice on x. In other words, instead of running Alice, we can approximate the protocol's result (referee's output) by a uniform sample from the multiset. Furthermore, we prove that such a multiset can be found (w.h.p.) by some (relatively few) independent samples from the distribution defined by Alice (given x).

Definition 3.4 (Characterizing Multiset)**.** For any r_{pub}, a multiset T of elements from M_A *characterizes* Alice for $x \in X$ if $\forall m_B \in M_B$,

$$\left| Q(T, m_B) - \Pr_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), m_B) = 1\right] \right| \leq 0.1$$

where $Q(T_x, m_B)$ is the referee's expected value for the multiset T_x and Bob's possible message $m_B \in M_B$ (Definition 3.3).

Lemma 3.5 (Sample a Characterizing Multiset)**.** *For any r_{pub}, for $x \in X$, let $r' = (r_A^1, ..., r_A^t)$ be t independent uniform samples from R_A where $t = 2 \cdot 200 \cdot \ln(2|M_B|)$. Then, for the multiset $T_x = \{A_{r_{pub}}(x, r_A^i) : i \in [t]\}$ it holds that $\forall m_B \in M_B$,*

$$\Pr_{r'}\left[\left| Q(T_x, m_B) - \Pr_{r_A}\left[\rho_{r_{pub}}(A_{r_{pub}}(x, r_A), m_B) = 1\right] \right| \leq 0.1\right] \geq 1 - \frac{1}{2|M_B|}$$

(i.e., T_x characterizes Alice for x)

Proof. Let T_x be as defined. $\forall i \in [t], m_B \in M_B$,

$$\mathbf{E}\left[\rho_{r_{\text{pub}}}(T_x[i], m_B)\right] = \Pr_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), m_B) = 1\right]$$
$$\implies \mathbf{E}\left[\rho_{r_{\text{pub}}}(T_x[i], m_B) - \Pr_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), m_B) = 1\right]\right] = 0$$

where the probability is over the random choice $T_x[i] \leftarrow A_{r_{\text{pub}}}(x)$.

Now, for $i \in [i]$, define random variables

$$\eta(i) = \rho_{r_{\text{pub}}}(T_x[i], m_B) - \Pr_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), m_B) = 1\right].$$

Since the members of T_x are independent random variables, we have that all $\{\eta(i) : i \in [t]\}$ are independent random variables with expectation 0. Hence, we can use a Chernoff bound to bound the probability that, for a fixed $m_B \in M_B$,

$$\left| \sum_{i \in [t]} \eta(T_x[i]) \right| > 0.1 \cdot t.$$

In other words, the probability that

$$\left| Q(T_x, m_B) - \Pr_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), m_B) = 1\right] \right| > 0.1$$

is bounded by

$$\Pr_{r'}[|\sum_{i\in[t]}\rho_{r_{\text{pub}}}(T_x[i],m_B)-\mathbf{E}[\sum_{i\in[t]}\rho_{r_{\text{pub}}}(T_x[i],m_B)]|>0.1]<2e^{-\frac{(0.1)^2\cdot t}{2}} \qquad \text{(Lemma 2.2)}$$
$$=2e^{-\frac{t}{200}}$$
$$=2e^{-2\ln(2|M_B|)}$$
$$=2\left(\frac{1}{2|M_B|}\right)^2$$
$$<\frac{1}{2|M_B|^2}$$

and by the union bound (over all $m_B \in M_B$),

$$\mathop{\mathbf{Pr}}_{r'}\left[\exists m_B \text{ s.t. } \left|Q(T,m_B)-\mathop{\mathbf{Pr}}_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x,r_A),m_B)=1\right]\right|>0.1\right]<|M_B|\cdot\frac{1}{2|M_B|^2}$$
$$=\frac{1}{2|M_B|}$$

□

We define a hash function by following the process of Lemma 3.5 (running Alice t times independently):

Construction 3.6 Characterizing Multiset Function

Definition: The function is defined by the public random r_{pub} and t Alice's random tapes $r_A^1,...,r_A^t \in R_A$.

Output: For $x \in X$, the value of the function is the multiset as in Lemma 3.5:

$$h(x) = \text{The multiset } \{A_{r_{\text{pub}}}(x,r_A^i) : i \in [t]\}$$

where the multiset is encoded as a sequence $A_{r_{\text{pub}}}(x,r_A^1),\ldots,A_{r_{\text{pub}}}(x,r_A^t)$, note that every Alice's message can be encoded using $\log|M_A| = c$ bits.

Observation 3.7. *For all $x \in X$, the function from Construction 3.6 outputs a multiset that characterizes x w.p. $1-\frac{1}{2|M_B|}$ where the probability is over the uniform random choice of $r_A^1,...,r_A^t \in R_A$.*

Observation 3.8. *The function from Construction 3.6 is compressing: The domain of the function is of size 2^n, but the range is of size at most*

$$(2^c)^t = 2^{400c\cdot(c+1)\cdot\ln 2} = 2^{\Theta(c^2)} = 2^{o(n)}$$

Next, we prove that any x and x' which share a characterizing multiset, induce bad inputs for the protocol (since Alice's behavior on x and x' is similar).

Claim 3.9. *Let $x, x' \in X$ and $y \in Y$ that separates them (Definition 2.4), if there is a multiset T that is characterizing for both x and x' then, the sum of the failure probability of $\pi(x, y)$ and $\pi(x', y)$ is at least 0.8. In other words, at least one of them fails.*

Proof. Since T is a characterizing multiset (Definition 3.4) of both x and x', then $\forall m_B \in M_B$

$$\left| Q(T, m_B) - \mathop{\mathbf{Pr}}_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), m_B) = 1\right] \right| \leq 0.1$$

and the same for x'. This means

$$\mathop{\mathbf{Pr}}_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), m_B) = 1\right] \in [Q(T, m_B) \pm 0.1]$$

and

$$\mathop{\mathbf{Pr}}_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x', r_A), m_B) = 1\right] \in [Q(T, m_B) \pm 0.1].$$

Putting it together we get that:

$$\left| \mathop{\mathbf{Pr}}_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), m_B) = 1\right] - \mathop{\mathbf{Pr}}_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x', r_A), m_B) = 1\right] \right| \leq 0.2. \tag{1}$$

Assume without loss of generality that $f(x, y) = 0$ and $f(x', y) = 1$

$$\begin{aligned}
\mathbf{Pr}[\pi \text{ fails on } (x, y)] &= \mathop{\mathbf{Pr}}_{r_A, r_B}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), B_{r_{\text{pub}}}(y, r_B)) = 1\right] \\
&= \mathop{\mathbf{E}}_{r_A, r_B}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), B_{r_{\text{pub}}}(y, r_B))\right] \\
&= \mathop{\mathbf{E}}_{r_B}\left[\mathop{\mathbf{E}}_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x, r_A), B_{r_{\text{pub}}}(y, r_B))\right]\right] \\
&\geq \mathop{\mathbf{E}}_{r_B}\left[\mathop{\mathbf{E}}_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x', r_A), B_{r_{\text{pub}}}(y, r_B))\right] - 0.2\right] && \text{(Equation (1))} \\
&= \mathop{\mathbf{E}}_{r_B}\left[\mathop{\mathbf{E}}_{r_A}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x', r_A), B_{r_{\text{pub}}}(y, r_B))\right]\right] - 0.2 \\
&= \mathop{\mathbf{Pr}}_{r_A, r_B}\left[\rho_{r_{\text{pub}}}(A_{r_{\text{pub}}}(x', r_A), B_{r_{\text{pub}}}(y, r_B)) = 1\right] - 0.2 \\
&= \mathbf{Pr}[\pi \text{ succeeds on } (x', y)] - 0.2 \\
&= 1 - \mathbf{Pr}[\pi \text{ fails on } (x', y)] - 0.2 \\
&= 0.8 - \mathbf{Pr}[\pi \text{ fails on } (x', y)]
\end{aligned}$$

Hence, the sum of the failure probability of the protocol on (x, y) and the failure probability of the protocol on (x', y) is

$$\mathbf{Pr}[\pi(x, y) \text{ fails}] + \mathbf{Pr}[\pi(x', y) \text{ fails}] \geq 0.8 \qquad \square$$

However, now we deal with the fact that there exist x's s.t. the multiset $h(x)$ does not characterize x (Observation 3.7).

Lemma 3.10. *Let π be an SM protocol of complexity $c(n) = o(\sqrt{n})$ and $h(x)$ be as in Construction 3.6. If we have an efficient adversary $\mathcal{A}\text{dv}_{collision}$ that breaks the security of h as a distributional CRH for some $p \in \text{poly}(\lambda)$:*

$$\Delta\left(\mathcal{A}\text{dv}_{collision}(h), \mathcal{COL}(h)\right) \leq \frac{1}{p(\lambda)}$$

Then, we can construct an adversary $\mathcal{A}\text{dv}_\pi$ with running time of the same order as $\mathcal{A}\text{dv}_{collision}$ s.t.

$$\mathbf{Pr}\left[\pi \textit{ fails on inputs from } \mathcal{A}\text{dv}_\pi\right] \geq 0.4\left(1 - \frac{1}{p(\lambda)}\right) - \text{negl}(\lambda)$$

Proof. $\mathcal{A}\text{dv}_\pi$'s algorithm is:

Algorithm 3.11 Near Ideal Collision Finder for h to Bad Inputs for Protocol π

1. Construct $h(x)$ using the public random string of π and as in Construction 3.6.
2. $x, x' \leftarrow \mathcal{A}\text{dv}_{\text{collision}}(h)$.
3. Find $y \in Y$ which separates x and x' (promised to be efficient by Definition 2.5).
4. Pass to Alice and Bob (x, y) w.p. $1/2$ or (x', y) w.p. $1/2$.

First, we consider $\mathcal{COL}$'s distribution: A pair (x, x') that was sampled from $\mathcal{COL}$ (the ideal collisions finder, Definition 2.12) will not be usable for Algorithm 3.11 if any of the following conditions hold:

1. $x = x'$.
2. $h(x) = h(x')$ is not characterizing x or x'.

We call a pair (x, x') a *colliding* pair if neither of the above two conditions hold. In the following claims we bound the probability for those bad events.

Claim 3.12. *The probability of sampling a pair (x, x) from $\mathcal{COL}$ (i.e., $x = x'$) is negligible. That is,*

$$\mathop{\mathbf{Pr}}_{(x,x')\leftarrow\mathcal{COL}}\left[x = x'\right] = \text{negl}(n)$$

Proof. First, consider the number of pairs (x, x') s.t. $x \neq x'$ but $h(x) = h(x')$. By the pigeonhole principle there exists a set of x's of size at least 2^{n-c^2} with the same image. Hence, there are at least $\binom{2^{n-c^2}}{2} = \Theta((2^{n-c^2})^2)$ many pairs (x, x') s.t. $x \neq x'$ but $h(x) = h(x')$. On the other hand, the number of pairs (x, x) is 2^n. Hence,

$$\mathop{\mathbf{Pr}}_{(x,x')\leftarrow\mathcal{COL}}\left[x = x'\right] = O\left(\frac{2^n}{2^n + (2^{n-o(n)})^2}\right) = \text{negl}(\lambda) \qquad \left(c^2 = o(n)\right)$$

$\square$

Claim 3.13. *The probability of sampling from $\mathcal{COL}$ a pair (x, x') s.t. the multiset $h(x)$ does not characterize x or x' is negligible.*

Proof. Let $(x, x') \leftarrow \mathcal{COL}$, recall that the distribution of each element from $\mathcal{COL}$ (x and x') is uniform (Definition 2.12). For each element in the pair, the probability that the multiset $h(x)$ does not characterize it is 2^{-c} and by the union bound the claim follows. □

By Claims 3.12 and 3.13, a sample from $\mathcal{COL}$ is colliding w.p. $1 - \text{negl}(\lambda)$. However, the distribution of $\mathcal{A}\text{dv}_{\text{collision}}$ is not exactly the same as $\mathcal{COL}$, but

$$\begin{aligned}\frac{1}{p(\lambda)} &\geq \Delta(\mathcal{COL}, \mathcal{A}\text{dv}_{\text{collision}}) \\ &\geq \left| \mathop{\mathbf{Pr}}_{(x,x') \leftarrow \mathcal{A}\text{dv}_{\text{collision}}} [(x, x') \text{ isn't colliding}] - \mathop{\mathbf{Pr}}_{(x,x') \leftarrow \mathcal{COL}} [(x, x') \text{ isn't colliding}] \right|\end{aligned}$$

and we can conclude that the probability that Algorithm 3.11 does not get a colliding pair (x, x') in step 2 is bounded by,

$$\mathop{\mathbf{Pr}}_{(x,x') \leftarrow \mathcal{A}\text{dv}_{\text{collision}}} [(x, x') \text{ isn't colliding}] \leq \frac{1}{p(\lambda)} + \text{negl}(\lambda)$$

To conclude: In cases that a colliding pair (x, x') was found by the adversary. The adversary chooses at random a pair from (x, y) and (x', y) (where y separates x and x', and can be found efficiently by Definition 2.5). By Claim 3.9,

$$\mathbf{Pr}\left[\pi(x, y) \text{ fails}\right] + \mathbf{Pr}\left[\pi(x', y) \text{ fails}\right] \geq 0.8$$

and hence the failure probability over the random choice of the pair is at least

$$\mathop{\mathbf{Pr}}_{(z,y) \underset{\pi}{\leftarrow} \mathcal{A}\text{dv}_\pi} [\pi(z, y) \text{ fails}] \geq 0.4$$

Now, put it together with the probability of finding a colliding pair (for h) and we get the probability that the protocol π fails on inputs from the adversary:

$$\begin{aligned}&\mathbf{Pr}\left[\mathcal{A}\text{dv}_\pi \text{ finds a colliding } (x, x')\right] \cdot \mathop{\mathbf{Pr}}_{(z,y) \underset{\pi}{\leftarrow} \mathcal{A}\text{dv}_\pi} [\pi(z, y) \text{ fails}] \\ &\geq \left(1 - \frac{1}{p(\lambda)} - \text{negl}(\lambda)\right) \cdot \frac{4}{10}\end{aligned}$$

□

We get that given an adversary for the distributional CRH we can find bad inputs for the protocol as required for the proof.

□

Interactive Protocols

For general (interactive) protocols we can also prove it for a logarithmic bound by the technique adaptation of Ben-Sasson and Maor [BM15]:

Theorem 3.14. *Let $\Omega(\log \lambda) \leq c(n) < \delta_\varepsilon \log n$, where δ_ε is a constant that depends only on ε. For an efficiently separable predicate (satisfying Definition 2.5), given a protocol of complexity $c(n)$ in the preset public coins interactive model, a distributional CRH can be constructed. (See Appendix A for the calculations of the constant δ_ε).*

Proof. Ben-Sasson and Maor studied protocols in the general communication settings and instead of using a characterizing multiset of *messages* they use a characterizing multiset of *deterministic strategies*. They have a variation of [BK97, Lemma 2.3] that says that there exists a strategies multiset of size $2^{O(c)}$ that characterizes the behavior of Alice for $x \in X$.

In the same way that Lemma 3.5 repeated the lemma of Babai and Kimmel with a larger constant, their lemma can be repeated to show that such a multiset can be sampled w.h.p. Note, the size of the set of possible strategies $|S_A|$ and $|S_B|$ is $O(2^{2^c})$

Lemma 3.15. *For $x \in X$, let $r' = (r_A^1, ..., r_A^t)$ be t independent uniform samples from R_A where $t = 2 \cdot 200 \cdot \ln(2|M_B|)$. Then, for the multiset $T_x = \{A_{r_{pub}}(x, r_A^i) : i \in [t]\}$ it holds that $\forall m_B \in S_B$,*

$$\Pr_{r'}\left[\left|Q(T_x, m_B) - \Pr_{r_A}\left[\rho_{r_{pub}}(A_{r_{pub}}(x, r_A), m_B) = 1\right]\right| \leq 0.1\right] \geq 1 - \frac{1}{2|S_B|}$$

Proof Sketch. As Lemma 3.5, $\forall m_B \in S_B$:

$$\Pr_{r'}[|\sum_{i\in[t]} \rho_{r_{\text{pub}}}(T_x[i], m_B) - \mathbf{E}[\sum_{i\in[t]} \rho_{r_{\text{pub}}}(T_x[i], m_B)]| > 0.1t] < 2e^{-\frac{(0.1)^2 \cdot t}{2}} \qquad \text{(Chernoff)}$$

$$< \frac{1}{2|S_B|^2}$$

and hence that probability that there exists such a bad m_B is, by the union bound, less than $\frac{1}{2|S_B|}$. □

Now, in the same way, Ben-Sasson and Maor observed that the number of possible strategies is at most $2^{2^{O(c)}}$ and deduced that for any private coins protocol $c \geq \delta_\varepsilon \log n$. Hence if $c < \delta_\varepsilon \log n$ there exist inputs x and x' with the same characterizing strategies multiset. Now we can repeat Lemma 3.10 to prove that an adversary $\mathcal{A}\text{dv}_{\text{collision}}$ capable of sampling 'near uniform' collisions can be used to find inputs (x, x') which make Alice 'behave similarly'. Let $y \in Y$ be as promised by Definition 2.4, we know that the protocol cannot be correct for both (x, y) and (x', y) (recall Claim 3.9). This means the protocol will fail on the adversarial input w.p. $> 1/3$. □

3.2.1 A Corollary for PPHs

Corollary 3.16. *Without assuming the existence of distributional CRHs one cannot get better than $\sqrt{\cdot}$ compression for a direct-access robust equality PPH, even when extending the definitions for randomized hash functions.*

Proof. Observe that any PPH can be used to solve the same problem in the preset public coins SM model. Hence, this corollary is simply rephrasing Theorem 3.2 in the terms of adversarially robust property-preserving hash functions. $\square$

Observe that the other direction can be done as well: Every preset public coins SM model protocol for $f(x, y)$ of $c(n)$ bits 'induces' a PPH for f of $c(n) \cdot n^{\varepsilon}$ bits for some $\varepsilon > 0$. That is, we can get any preset public coins SM protocol, repeat it n^{ε} times to make the error probability negligible. This protocol is a family of PPHs and its random coins are fixed when sampling a function from the family.

4 No Ultra Short Interactive Communication

The power of the preset public coins model power lies between the public and the private coins model. As noted, the public random coins model is strictly more powerful than the private one: there are protocols of $O(1)$ bits only in this model. We show that in our model there are no functions with $o(\log \log n)$ communication:

Theorem 4.1. *Let $c(n) \colon \mathcal{N} \mapsto \mathcal{N}$ be s.t. $2^{3c(n)} = O(\log n)$ and let $f \colon X \times Y \to \{0, 1\}$ be an efficiently separable predicate (satisfying Definition 2.5, i.e., non redundant s.t. can be proven efficiently). In the preset public coins interactive communication model, if the adversary has a running time of $\mathrm{poly}(\lambda)$ then, there are no protocols of complexity $O(c(n))$. (See Appendix A for tighter constants).*

Proof. Assume there is such a protocol in the preset public coins interactive model for some non-redundant function f of complexity $c(n)$.

In the proof of Theorem 3.14 we adapted Construction 3.6 for interactive protocols. The constructed hash function has the following properties:

- Random collisions in the function induce (w.h.p.) bad inputs in the protocol (Lemma 3.10).
- The range of the function is of size

$$|S_A|^t = |S_A|^{2^{2 \cdot 200 \ln(2|S_B|)}}$$

Those properties are the key points of the adversary described by Algorithm 4.2 that search for random collisions by a brute force search.

Algorithm 4.2 Finding Bad Inputs in Ulta-Succinct Protocols

1. Construct a characterizing function $h(\cdot)$ (Construction 3.6).
2. Repeat at most $3 \cdot 2^{2^{3c}} = \text{poly}(\lambda)$ times:
   (a) Choose $x \neq x' \in X$ uniformly at random.
   (b) If $h(x) = h(x')$:
      i. Find $y \in Y$ that separates x and x' (can be done efficiently as promised by Definition 2.5).
      ii. Output (x, y) w.p. $1/2$ or (x', y) w.p. $1/2$
      iii. Halt

Let h be a characterizing function of the protocol (Construction 3.6). The proof relies on the following two claims:

Claim 4.3. *There must be a collision in h.*

Proof. The range of the characterizing function $h(x)$ is of size (number of possible characterizing sets):

$$|S_A|^{2^{2\cdot 200\ln(2|S_B|)}} = 2^{2^c\cdot 2\cdot 200\ln\left(2^{2^c}+1\right)} < 2^{2^{3c}}$$

Hence, since $2^{3c} = O(\log n) = o(n)$ there must be a collision in the function. $\square$

Claim 4.4. *The adversary described in Algorithm 4.2 finds a collision w.h.p.*

Proof. Because the range is small (same order as the running time of the adversary $2^{2^{3c}} = \text{poly}(\lambda)$), the adversary can find random collisions easily. The probability for a random pair to collide is at least $\frac{1}{2^{2^{3c}}}$ and hence, after $3 \cdot 2^{2^{3c}}$ tries, the probability that a collision was not found is at most:

$$\begin{aligned} &\Pr_{x,x'}\left[h(x) \neq h(x')\right]^{3\cdot 2^{2^{3c}}} \leq \left(\left(1 - \frac{1}{2^{2^{3c}}}\right)^{2^{2^{3c}}}\right)^3 \to e^{-3} \\ \implies &\Pr_{x,x'}\left[h(x) \neq h(x')\right]^{3\cdot 2^{2^{3c}}} < 0.05 \end{aligned}$$

$\square$

We get that w.h.p. the adversary finds a collision in the function. However, not every collision implies bad inputs for the protocol: The construction of the characterizing function implies that there exist also bad collisions: x and x' s.t. $h(x) = h(x')$ but $h(x)$ doesn't characterizes x or x' (recall Observation 3.7). However, in almost all collisions it is not the case and $h(x)$ characterizes x and x' (recall Claim 3.13). Now, since the collision that Algorithm 4.2 finds is completely random we can conclude,

$$\Pr\left[\text{the adversary will find a colliding pair}\right] \geq 1 - 0.05 - \frac{1}{|S_B|}$$

and by Claim 3.9

$$\mathbf{Pr}\left[\text{the protocol will fail}\right] \geq \frac{1}{2} \cdot \frac{8}{10}\left(1 - 0.05 - \frac{1}{|S_B|}\right) > \frac{1}{3}$$

□

5 Secret Key Agreement from Efficient SM Protocols

5.1 Optimal Protocols from SKA

Our first observation is that it is possible to obtain an optimal protocol (in terms of the error as a function of the communication) for the equality predicate once given a secret key agreement protocol, following relatively simple principles. The error is 2^{-c} (where c is the communication complexity after the free talk) plus a negligible factor reflecting the probability of breaking the secret-key exchange. For completeness, we give full details in Appendix B.1.

Theorem 5.1. *In the stateful preset public coins SM with free talk model: Given a secret key agreement protocol there is, for any $c(n)$, a protocol for the equality predicate of complexity $c(n)$, where any adversary can cause an incorrect answer with probability at most $2^{-c} + \text{negl}(n)$.*

5.2 SKA from Near Optimal Protocols

Theorem 5.2. *An SM protocol with stateful free talk for the equality predicate of complexity $c(n) = O(\log \log n)$ for $c(n)$ larger from some constant, with failure probability $\varepsilon \leq 2^{-0.7c(n)}$ that is secure against a rushing adversary, implies the existence of secret key-agreement protocols.*

Proof. Assume we have such a protocol π for the equality predicate $EQ\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$. We will use π for constructing a secret key-agreement protocol. The idea is to construct a weak secret bit agreement that can be amplified to a full secret key agreement (α and β according to Theorem 2.16): The construction is based on the following (α, β)-SBA protocol:

Algorithm 5.3 Weak Bit Agreement

1. Alice and Bob communicate and toss coins according to the free talk of protocol π to generate their secret states τ_A and τ_B respectively.

2. Alice selects at random a bit $b \in \{0,1\}$ and uniformly random inputs $x_0, x_1 \in \{0,1\}^n$.

3. Alice evaluates $m_A = A(x_b, \tau_A)$ (that is, a message of the protocol π for $EQ(\cdot,\cdot)$).

4. Alice sends to Bob (m_A, x_1).

5. Bob evaluates $m_B = B(x_1, \tau_B)$.

6. Alice outputs b and Bob outputs $b' = \rho(m_A, m_B)$.

Lemma 5.4. *Algorithm 5.3 is a* $\left(1 - 2^{-c/2-3}, 2^{-c/2+1}\right)$*-SBA protocol.*

Proof. Let $c = c(n)$ and $\delta = 2^{c/2}$. We have to show its agreement and secrecy properties:

Agreement. By the definition of the protocol π:

$$\mathbf{Pr}\left[b = b'\right] \geq 1 - \left(\frac{1}{2}\right)^{0.7c} \geq 1 - \left(\frac{1}{2}\right)^{0.5c-2} = \frac{1 + (1 - 2^{-c/2-3})}{2} = \frac{1+\alpha}{2}$$

Secrecy. We should show that for every PPTM adversary $\mathcal{A}\mathrm{dv}_{\mathrm{sba}}$

$$\mathbf{Pr}\left[\mathcal{A}\mathrm{dv}_{\mathrm{sba}}(m_A, x_1) = b \mid b = b'\right] \leq \frac{2-\beta}{2} = \frac{2 - 2^{-c/2+1}}{2} = \frac{2^{c/2} - 1}{2^{c/2}} = \frac{\delta - 1}{\delta}$$

Assume towards contradiction that $\mathbf{Pr}\left[\mathcal{A}\mathrm{dv}_{\mathrm{sba}}(m_A, x_1) = b \mid b = b'\right] > \frac{\delta-1}{\delta}$. We show that given $\mathcal{A}\mathrm{dv}_{\mathrm{sba}}$, we can construct $\mathcal{A}\mathrm{dv}_{\mathrm{eq}}$ that finds bad inputs for the protocol π (with probability higher than ε):

Claim 5.5. *Given* $\mathcal{A}\mathrm{dv}_{sba}$ *with success probability at least* $\frac{\delta-1}{\delta}$*, we can construct an adversary* $\mathcal{A}\mathrm{dv}_{eq}$ *with running time* $O(2^{3c}) = O(\delta^6)$ *s.t.*

$$\mathbf{Pr}\left[\pi \textit{ fails on inputs from } \mathcal{A}\mathrm{dv}_{eq}\right] > 2^{-0.7c} \geq \varepsilon$$

Proof. The strategy of the adversary $\mathcal{A}\mathrm{dv}_{\mathrm{eq}}$ to find bad inputs is:

Algorithm 5.6 $\mathcal{A}\mathrm{dv}_{\mathrm{eq}}$ – Find Bad Inputs Using $\mathcal{A}\mathrm{dv}_{\mathrm{sba}}$

1. Select uniformly at random $x \in \{0,1\}^n$ and set it as Alice's input.
2. Let Alice's message (output) be $m_A \in M_A$.
3. Repeat at most 2^{3c} times:
   (a) Select uniformly at random $x' \in \{0,1\}^n$.
   (b) If $\mathcal{A}\mathrm{dv}_{\mathrm{sba}}(x', m_A) = 1$:
      i. Set Bob's input to be either $y = x$ w.p. $1/2$ or $y = x'$ w.p. $1/2$.
      ii. Halt.

Recall that the private states of Alice and Bob are τ_A and τ_B (unknown to the adversary). The success of the adversary $\mathcal{A}\mathrm{dv}_{\mathrm{eq}}$ relies on choosing a *colliding* x' (i.e., x' s.t. $B(x,\tau_B) = B(x',\tau_B)$). For any $x \in \{0,1\}^n$, let p_x be the probability that a random x' will collide with x. I.e.,

$$p_x = \frac{|\{x' : B(x',\tau_B) = B(x,\tau_B)\}|}{2^n}.$$

Note that $\mathbf{E}_x[p_x] \geq 2^{-c}$, since there are at most 2^c possible messages for Bob. Suppose that chosen a colliding x' in step 3(a) (i.e., $B(x',\tau_B) = B(x,\tau_B)$). Then, the adversary $\mathcal{A}\mathrm{dv}_{\mathrm{eq}}$ will identify (with some probability) that this is the case by checking whether $\mathcal{A}\mathrm{dv}_{\mathrm{sba}}(x', m_A) \stackrel{?}{=} 1$ (step 3(b)).

If not a colliding x' was chosen, then the adversary $\mathcal{A}\mathrm{dv}_{\mathrm{eq}}$ can identify it using $\mathcal{A}\mathrm{dv}_{\mathrm{sba}}$ and retry. That is:

1. In every try: The adversary chooses a random a colliding $x' \in \{0,1\}^n$ w.p. at least $p_x \cdot \frac{\delta-1}{\delta}$.
2. For the i-th try: The probability that the previous x''s were not colliding but the adversary identified it, is

$$\left((1-p_x)\cdot\frac{\delta-1}{\delta}\right)^{i-1}.$$

This means,

$$
\begin{aligned}
\Pr_{x' \leftarrow \mathcal{A}\mathrm{dv}_{\mathrm{eq}}}[B(x', \tau_B) = B(x, \tau_B)] &= \sum_{i=0}^{\delta^6-1} \mathbf{Pr}\,[\text{Success in the } (i+1)\text{-th try}] \\
&\quad \cdot \mathbf{Pr}\,[\text{The previous } i \ x'\text{'s were not colliding but identified}] \\
&\geq \sum_{i=0}^{\delta^6-1} p_x \cdot \frac{\delta-1}{\delta} \left((1-p_x) \frac{\delta-1}{\delta} \right)^i \\
&= p_x \cdot \frac{\delta-1}{\delta} \cdot \frac{1 - \left((1-p_x)\frac{\delta-1}{\delta} \right)^{\delta^6}}{1-(1-p_x)\frac{\delta-1}{\delta}} \\
&> \frac{p_x \cdot \frac{\delta-1}{\delta} \cdot 0.99}{1-(1-p_x)\frac{\delta-1}{\delta}} \\
&= \frac{0.99 \cdot p_x(\delta-1)}{\delta-(1-p_x)(\delta-1)} \\
&= \frac{0.99 \cdot p_x(\delta-1)}{1+p_x(\delta-1)}.
\end{aligned}
$$

To complete the proof we have to show that

$$\frac{0.99 \cdot p_x(\delta-1)}{1+p_x(\delta-1)} \geq 2^{-0.7c}$$

over the random choice of x (step (1)). We consider when

$$\frac{0.99 \cdot p_x(\delta-1)}{1+p_x(\delta-1)} \geq p_x \cdot 2^{0.3c+1} \tag{2}$$

it is equivalent to

$$p_x \leq \frac{0.99(\delta-1) - 2^{0.3c+1}}{2^{0.3c}(\delta-1)}$$

but since $\frac{0.99(\delta-1)-2^{0.3c+1}}{2^{0.3c+1}(\delta-1)} = \frac{0.99(2^{c/2}-1)-2^{0.3c+1}}{2^{0.3c+1}(2^{c/2}-1)} \geq 2^{-0.3c-2}$ it is sufficient to show that

$$p_x < 2^{-0.3c-2}. \tag{3}$$

Now, we argue that Equation (3) holds for all x. But first we have to prove the following claim:

Claim 5.7. *Let π be a protocol for the equality predicate with stateful free talk. Let τ_B be the secret state of Bob, let $x, y \in \{0,1\}^n$ s.t. $x \neq y$ and $B(x, \tau_B) = B(y, \tau_B)$ then, for $z \in_R \{x, y\}$,*

$$\Pr_{\substack{\tau_A, \tau_B \\ x,y \\ z \in \{x,y\}}} [\pi(x, z) \text{ fails}] = \frac{1}{2}$$

Proof. Since Bob will send the same message for both x and y this means that the output of the protocol will be the same. Therefore, the failure probability of the protocol in both cases will sum to 1:

$$\begin{aligned}
\Pr_{\tau_A,\tau_B,x,y}[\pi(x,y) \text{ fails}] &= \Pr_{\tau_A,\tau_B,x,y}[\rho(A(x,\tau_A),B(y,\tau_B)) = 1] \\
&= \Pr_{\tau_A,\tau_B,x,y}[\rho(A(x,\tau_A),B(x,\tau_B)) = 1] \\
&= 1 - \Pr_{\tau_A,\tau_B,x}[\rho(A(x,\tau_A),B(x,\tau_B)) = 0] \\
&= 1 - \Pr_{\tau_A,\tau_B,x}[\pi(x,x) \text{ fails}] \\
\implies \Pr_{\tau_A,\tau_B,x,y}[\pi(x,y) \text{ fails}] + \Pr_{\tau_A,\tau_B,x}[\pi(x,x) \text{ fails}] &= 1
\end{aligned}$$

hence,

$$\Pr_{\substack{\tau_A,\tau_B \\ x,y \\ z\in\{x,y\}}}[\pi(x,z) \text{ fails}] = \frac{1}{2}\cdot(1) = \frac{1}{2}.$$

$\square$

Now we can prove that Equation (3) must hold:

Claim 5.8. *For all secret states of Bob τ_B, $\forall x \in \{0,1\}^n$*

$$p_x < 2^{-0.3c-2}$$

Proof. Assume that there exists $w \in \{0,1\}^n$ s.t. $p_w \geq 2^{-0.3c-2}$. This implies a too high probability that Bob will send the same message for random inputs (i.e., the probability for a collision is 'too high'):

$$\begin{aligned}
&\Pr_{x,x'}[B(x,\tau_B) = B(x',\tau_B)] \geq \left(2^{-0.3c-2}\right)^2 \\
\implies &\Pr_{\substack{x,x' \\ z\in\{x,x'\}}}[\pi(x,z) \text{ fails}] \geq 2^{-0.6c-4} && \text{(Claim 5.7)}
\end{aligned}$$

and for large enough c we get a contradiction since $\varepsilon = 2^{-0.7c} < 2^{-0.6c-4}$. $\square$

By Claim 5.8 we get that Equation (2) holds and hence

$$\Pr_{y\leftarrow\mathcal{A}\mathrm{dv_{eq}}}[B(y,\tau_B) = B(x,\tau_B)] > p_x \cdot 2^{0.3c+1}$$

this means that over the random choice of x:

$$\mathop{\mathbf{E}}_{x}\left[\Pr_{y\leftarrow\mathcal{A}\mathrm{dv_{eq}}}[B(y,\tau_B) = B(x,\tau_B)]\right] > 2^{-c}\cdot 2^{-0.3c+1} = 2^{-0.7c+1}$$

and we can conclude by Claim 5.7 that on the inputs (x,y) from $\mathcal{A}\mathrm{dv_{eq}}$,

$$\Pr[\pi(x,y) \text{ fails}] > 2^{-0.7c+1}\cdot\frac{1}{2} = 2^{-0.7c}$$

$\square$

Claim 5.5 implies the secrecy of Algorithm 5.3 (otherwise, we get a contradiction for the security of protocol π). This means Algorithm 5.3 is an $(1-2^{-c/2-3}, 2^{-c/2+1})$-SBA. $\square$

Finally, we show that Theorem 2.16 can be used:

Claim 5.9. *For the functions $\alpha(c) = 1 - 2^{-c/2-3}$ and $\beta(c) = 2^{-c/2+1}$ the conditions in Theorem 2.16 hold.*

Proof. In Theorem 2.16 there are 2 conditions:

1. $\frac{1-\alpha}{1+\alpha} < \beta$ holds since,

$$\frac{1-(1-2^{-c/2-3})}{1+(1-2^{-c/2-3})} = \frac{2^{-c/2-3}}{2-2^{-c/2-3}} = \frac{1}{2^{c/2+4}-1} < \frac{1}{2^{c/2-1}}$$

2. For showing the second condition we use the following fact:

   Fact 5.10. *For $0 \le x \le 1$,*

$$x \le \log(1+x)$$

   Now, we calculate φ:

$$\begin{aligned} &\beta \cdot \frac{1+\alpha}{1-\alpha} = 2^{-c/2+1} \cdot (2^{c/2+4}-1) > 2^4 \\ \implies &\log\left(\beta \cdot \frac{1+\alpha}{1-\alpha}\right) > 4 \\ \implies &\frac{8}{\log\left(\beta \cdot \frac{1+\alpha}{1-\alpha}\right)} < \frac{8}{4} \\ \implies &\varphi = \max\left(2, \frac{8}{\log\left(\beta \cdot \frac{1+\alpha}{1-\alpha}\right)}\right) = 2 \end{aligned}$$

   and bound γ,

$$\begin{aligned} &\left(\frac{1-\alpha}{1+\alpha}\right)^{\varphi} = \left(\frac{1}{2^{c/2+4}-1}\right)^2 > 2^{-c-8} \\ \implies &\log\left(1+\left(\frac{1-\alpha}{1+\alpha}\right)^{\varphi}\right) > 2^{-c-8} && \text{(Fact 5.10)} \\ \implies &\gamma = \frac{1}{\log\left(1+\left(\frac{1-\alpha}{1+\alpha}\right)^{\varphi}\right)} < 2^{c+8} \end{aligned}$$

   Hence, the second condition also holds:

$$\frac{\varphi 2^{4\gamma}}{\alpha} \le \frac{2 \cdot 2^{4 \cdot 2^{c+8}}}{1-2^{-c/2-3}} = O\left(2^{4 \cdot 2^{\log\log n + 8}}\right) = \mathrm{poly}(\lambda)$$

$\square$

We conclude, by Claim 5.9 that the SBA of Algorithm 5.3 (Lemma 5.4) can be amplified efficiently to a secret key agreement protocol. $\square$

6 Conclusions

Role of private randomness. In this paper we introduced a computational model for communication complexity. However, it can also be seen as a generalization of (deterministic) property preserving hash functions to probabilistic algorithms. We studied some relations between the power of private randomness and cryptographic primitives such as collision resistance. The main open problem left from this point is whether CRHs are equivalent to preset public coins SM protocols of complexity $o(\sqrt{n})$ and whether we can break that bound using a primitive weaker than CRHs. Another direction could be to show how to use $o(\sqrt{n})$ equality protocols in order to get low communication string commitment.

Boyle et al.'s lower bounds. Boyle et al. [BLV18] proved two general lower bounds for property preserving hash functions using communication complexity[10]:

1. A lower bound for *reconstructing* predicates: Boyle et al. proved that for predicates that can be used for reconstructing the original string there cannot exist (compressing) property preserving hash functions. This lower bound is also true for our preset public coins SM model. However, we didn't necessarily consider reconstructing predicates (for instance, the equality predicate is not a reconstructing predicate).

2. General lower bound from one-way communication: Boyle et al. proved that any property preserving hash function cannot compress better than the one-way communication complexity[11]. This lower bound is also true in our model, but it is too loose in our context since in our model the inputs and the public random string may be dependent (e.g., the equality predicate complexity is $O(1)$ in the one-way communication complexity model).

Multi CRHs (MCRH). For $k \geq 3$, k-multi-collision resistance is where finding a collision of size k is hard: Any PPTM can find $x_1, \ldots, x_k$ s.t. $h(x_1) = \ldots = h(x_k)$ with at most negligible probability (for $k = 2$ it is the regular notion of collision resistance). It is known that there does not exist a black-box construction of dCRHs using MCRHs[12] and since we showed a construction of dCRHs from protocols with some small communication complexity, we conclude that MCRHs cannot be used alone in a black-box manner to achieve such small communication complexity.

Secret key agreement. We showed a tight relationship between secret key agreement protocols and succinct protocols for the equality predicate in the SM preset public coins stateful free talk model. On the one hand, SKA can be used for constructing an equality protocol in this model, and on the other hand, equality protocols with good error in this model can be used for constructing SKA protocols. The open question is whether the

[10]See also Hardt and Woodruff [HW13] who proved robustness limitations for *linear* functions.

[11]See also Fleischhacker and Simkin [FS21] and Fleischhacker et al. [FLS22] for more such lower bounds.

[12]Komargodski et al. [KNY18] proved a black-box separation of MCRHs from CRHs. Komargodski and Yogev [KY18] observed that it also holds for dCRHs since it uses the same collision finder of Simon [Sim98]. However, Komargodski and Yogev showed a *non-black-box* construction of dCRHs using MCRHs.

existence of protocols with much worse error probability (e.g., constant error probability for c which $O(\log\log\lambda)$) also imply SKA.

Acknowledgments

We thank Shahar Dobzinski, Guy Rothlbum and Eylon Yogev for useful discussions and suggestions.

Bibliography

[AS08] Noga Alon and Joel H. Spencer. *The Probabilistic Method, Third Edition*. Wiley, 2008.

[BGK15] Ralph Bottesch, Dmitry Gavinsky, and Hartmut Klauck. “Equality, revisited”. In: *International Symposium on Mathematical Foundations of Computer Science*. Springer. 2015, pp. 127–138.

[BHKY19] Nir Bitansky, Iftach Haitner, Ilan Komargodski, and Eylon Yogev. “Distributional collision resistance beyond one-way functions”. In: *EUROCRYPT 2019*. Springer. 2019, pp. 667–695.

[BK97] László Babai and Peter G Kimmel. “Randomized simultaneous messages: Solution of a problem of Yao in communication complexity”. In: *Proceedings of Computational Complexity. Twelfth Annual IEEE Conference*. IEEE. 1997, pp. 239–246.

[BLV18] Elette Boyle, Rio LaVigne, and Vinod Vaikuntanathan. “Adversarially Robust Property-Preserving Hash Functions”. In: *ITCS 2019*. 2018.

[BM09] Boaz Barak and Mohammad Mahmoody-Ghidary. “Merkle puzzles are optimal—an $O(n^2)$-query attack on any key exchange from a random oracle”. In: *CRYPTO 2009*. Springer. 2009, pp. 374–390.

[BM15] Eli Ben-Sasson and Gal Maor. “Lower bound for communication complexity with no public randomness”. In: *Electron. Colloquium Comput. Complex.* 22 (2015), p. 139.

[DI06] Bella Dubrov and Yuval Ishai. “On the randomness complexity of efficient sampling”. In: *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*. 2006, pp. 711–720.

[FLS22] Nils Fleischhacker, Kasper Green Larsen, and Mark Simkin. “Property-Preserving Hash Functions from Standard Assumptions”. In: *EUROCRYPT 2022*. 2022. to appear.

[FS21] Nils Fleischhacker and Mark Simkin. “Robust Property-Preserving Hash Functions for Hamming Distance and More”. In: *EUROCRYPT 2021*. Lecture Notes in Computer Science. Springer, 2021.

[GHY20] Ofer Grossman, Justin Holmgren, and Eylon Yogev. "Transparent Error Correcting in a Computationally Bounded World". In: *TCC 2020*. Springer, 2020.

[HIKNV04] Prahladh Harsha, Yuval Ishai, Joe Kilian, Kobbi Nissim, and Srinivasan Venkatesh. "Communication versus computation". In: *International Colloquium on Automata, Languages, and Programming*. Springer. 2004, pp. 745–756.

[Hol05] Thomas Holenstein. "Key agreement from weak bit agreement". In: *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*. 2005, pp. 664–673.

[Hol06] Thomas Holenstein. "Strengthening key agreement using hard-core sets". PhD thesis. ETH Zurich, 2006.

[HW13] Moritz Hardt and David P Woodruff. "How robust are linear sketches to adaptive inputs?" In: *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*. 2013, pp. 121–130.

[IL89] Russell Impagliazzo and Michael Luby. "One-way functions are essential for complexity based cryptography". In: *30th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society. 1989, pp. 230–235.

[IR89] Russell Impagliazzo and Steven Rudich. "Limits on the provable consequences of one-way permutations". In: *Proceedings of the twenty-first annual ACM symposium on Theory of computing*. 1989, pp. 44–61.

[KN96] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, 1996.

[KNY18] Ilan Komargodski, Moni Naor, and Eylon Yogev. "Collision Resistant Hashing for Paranoids: Dealing with Multiple Collisions". In: *Advances in Cryptology - EUROCRYPT 2018 Proceedings, Part II*. Springer, 2018, pp. 162–194.

[KY18] Ilan Komargodski and Eylon Yogev. "On Distributional Collision Resistant Hashing". In: *Advances in Cryptology - CRYPTO 2018*. Vol. 10992. Lecture Notes in Computer Science. Springer, 2018, pp. 303–327.

[MNS11] Ilya Mironov, Moni Naor, and Gil Segev. "Sketching in adversarial environments". In: *SIAM Journal on Computing* 40.6 (2011), pp. 1845–1870.

[New91] Ilan Newman. "Private vs. Common Random Bits in Communication Complexity". In: *Inf. Process. Lett.* 39.2 (1991), pp. 67–71.

[NR09] Moni Naor and Guy N Rothblum. "The complexity of online memory checking". In: *Journal of the ACM (JACM)* 56.1 (2009), pp. 1–46.

[NS96] Ilan Newman and Mario Szegedy. "Public vs. private coin flips in one round communication games". In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. 1996, pp. 561–570.

[RY20] Anup Rao and Amir Yehudayoff. *Communication Complexity: and Applications*. Cambridge University Press, 2020.

[Sim98] Daniel R Simon. “Finding Collisions on a One-Way Street: Can Secure Hash Functions Be Based on General Assumptions?” In: *Advances in Cryptology - EUROCRYPT '98*. Springer. 1998, pp. 334–345.

[WC79] Mark N. Wegman and Larry Carter. “New Classes and Applications of Hash Functions”. In: *20th Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 29-31 October 1979*. IEEE Computer Society, 1979, pp. 175–182.

[Wee07] Hoeteck Wee. “One-Way Permutations, Interactive Hashing and Statistically Hiding Commitments”. In: *TCC 2007*. Ed. by Salil P. Vadhan. Vol. 4392. Lecture Notes in Computer Science. Springer, 2007, pp. 419–433.

[Yao79] Andrew Chi-Chih Yao. “Some Complexity Questions Related to Distributive Computing(Preliminary Report)”. In: STOC '79. Association for Computing Machinery, 1979, pp. 209–213.

A Calculating The Constants for Interactive Communication Lower Bound

Lemma A.1 ([BM15, Lemma 1])**.** *For every δ, for $t = \frac{2}{\delta^2}\ln(2|S_B|)$ there exists a characterizing multiset T of size t s.t. $\forall m_B \in M_B$:*

$$\left| Q(T, m_B) - \Pr_{r_A}\left[\rho(A(x, r_A), m_B)\right] \right| < \delta$$

By generalizing the proof of Lemma 3.10 we can claim that for x and x' with the same characterizing multiset T, $\forall m_B \in S_B$:

$$\begin{aligned}
&\begin{cases} \left| Q(T, m_B) - \Pr_{r_A}\left[\rho(A(x, r_A), m_B)\right] \right| < \delta \\ \left| Q(T, m_B) - \Pr_{r_A}\left[\rho(A(x', r_A), m_B)\right] \right| < \delta \end{cases} \\
\Longrightarrow & \left| \Pr_{r_A}\left[\rho(A(x, r_A), m_B)\right] - \Pr_{r_A}\left[\rho(A(x', r_A), m_B)\right] \right| < 2\delta \\
\Longrightarrow & \Pr\left[\pi(x, y) \text{ fails}\right] + \Pr\left[\pi(x', y) \text{ fails}\right] > 1 - 2\delta \\
\Longrightarrow & \Pr_{z \in \{x, x'\}}\left[\pi(z, y) \text{ fails}\right] > \frac{1}{2} - \delta
\end{aligned}$$

The protocol fails on such inputs if

$$\begin{aligned}
& \frac{1}{2} - \delta \geq \varepsilon \\
\Longleftrightarrow & \frac{1}{2} - \varepsilon \geq \delta
\end{aligned}$$

This means,

$$t \geq \frac{2}{(0.5 - \varepsilon)^2} \ln\left(2^{2^c}\right) = \frac{2^{c+1} \cdot \ln 2}{(0.5 - \varepsilon)^2}$$

and the number of possible multisets is at least

$$\begin{aligned}|S_A|^t &\geq \left(2^{2^c}\right)^{\frac{2^{c+1}\cdot\ln 2}{(0.5-\varepsilon)^2}}\\ &= 2^{2^c\cdot 2^{c+1}\cdot\ln 2/(0.5-\varepsilon)^2}\\ &= 2^{2^{2c+1}\cdot\ln 2/(0.5-\varepsilon)^2}\end{aligned}$$

The number of characterizing multisets is less than the number of inputs 2^n when

$$\begin{aligned}&2^n > |S_A|^t\\ \iff &\log n > 2c+1+\log\ln 2 - \log(0.5-\varepsilon)^2\\ \iff &\log n > 2c+1+\log\ln 2 - 2\left(\log(1-2\varepsilon)-1\right)\\ \iff &\log n > 2c+3+\log\ln 2 - 2\log(1-2\varepsilon)\\ \iff &c < \frac{1}{2}\log n - \frac{3}{2} - \frac{1}{2}\log\ln 2 + \log(1-2\varepsilon)\end{aligned}$$

We can conclude that in the interactive communication settings, for protocols of complexity $c(n)$ where

$$c < \frac{1}{2}\log n - O(1)$$

it holds that,

1. Cannot exists in the private coins model [BM15].
2. Cannot exists in the preset public coins model if the adversary has poly $\left(2^{O\left(2^{2c}\right)}\right)$ running time (Theorem 4.1).
3. If there exists in the preset public coins model then, it can be used to construct a dCRH (Theorem 3.14).

B Omitted Proofs

B.1 Proof of Theorem 5.1

We will use a family of pairwise independent hash functions:

Definition B.1 (Pairwise Independent Hash)**.** Let $\mathcal{K}$ be a set of keys (descriptions of function). A family of hash functions $\mathcal{H} = \{h_k\colon X \to Y : k \in \mathcal{K}\}$ is pairwise independent if $\forall x_1 \neq x_2 \in X$ and $\forall y_1, y_2 \in Y$,

$$\Pr_{k\in\mathcal{K}}\left[h_k(x_1) = y_1 \wedge h_k(x_2) = y_2\right] = \frac{1}{|Y|^2}$$

Fact B.2. *There are well known constructions of pairwise independent hash functions families that use* $O(\log |Y|)$ *random bits (i.e.,* $|k| = O(\log |Y|)$*).*

We prove Theorem 5.1 by constructing such a protocol using a secret key agreement protocol and pairwise independent hash functions.

Algorithm B.3 Optimal Equality Predicate Protocol Using a Secret Key Agreement

---

Let $c \in \mathbb{N}$ (it will be the communication complexity of the protocol).

1. Free talk: Alice and Bob agree on a secret $O(c)$ bits random string k (i.e., they preform an SKA protocol).
2. Alice receives her input $x \in \{0,1\}^n$.
3. Alice maps k to a pairwise independent hash function $h_k \colon \{0,1\}^n \to \{0,1\}^c$.
4. Alice sends to the referee $h_k(x)$.
5. Bob receives his input $y \in \{0,1\}^n$.
6. Bob map k to the same function h_k.
7. Bob sends to the referee $h_k(y)$.
8. The referee checks whether $h_k(y) \stackrel{?}{=} h_k(x)$ and outputs it as the result.

Claim. *For any* $c(n)$*, the failure probability of the protocol in Algorithm B.3 is bounded by* $2^{-c} + \mathrm{negl}(n)$ *while the communication complexity of the protocol (following the free talk) is* $c(n)$*.*

Proof. If $x = y$ then $h_k(x) = h_k(y)$ and the protocol will succeed. Let $x \neq y$ and assume first that k is completely random and unknown to the adversary. Since the function is pairwise independent, the value $h(y)$ for y chosen by any adversary will be random in the sense that $\forall z \in \{0,1\}^c$:

$$\begin{aligned}
\Pr_k \left[h_k(y) = z \mid h_k(x) = z\right] &= \frac{\mathbf{Pr}_k \left[h_k(y) = z \wedge h_k(x) = z\right]}{\mathbf{Pr}_k \left[h_k(x) = z\right]} \\
&= \Pr_k \left[h_k(y) = z\right] && \text{(pairwise ind.)} \\
&= 2^{-c}
\end{aligned}$$

Now, any adversary who can find bad inputs can be used for distinguishing k and a uniformly random string. But, from the security definition of secret key agreement, any bounded time adversary cannot distinguish with a non-negligible advantage between k and a uniformly random string. Hence, we can conclude that the failure probability of the protocol is bounded by $2^{-c} + \mathrm{negl}(n)$. ∎

Reducing the Free Talk. We note that in general the free talk can be short: One can use a pseudorandom generator on a small secret seed or use a method like in Wegman and Carter [WC79].

Note that any function that has an optimal protocol (in terms of error for the given communication complexity, i.e. 2^{-c}) in the public coins model can follow a transformation as described above and result in an optimal protocol in the stateful preset public coins SM with free talk model.

B.2 Proof of Theorem 3.1

In the SM model, recall that there exists a protocol of complexity $O(\sqrt{n})$ (Fact 2.8), we apply the protocol on the output of the CRH:

Algorithm B.4 Equality SM Protocol of Complexity $O(\sqrt{\lambda})$

---

1. Alice and Bob sample a common CRH $h\colon \{0,1\}^n \to \{0,1\}^\lambda$.
2. Alice and Bob apply h and compress their inputs to size λ.
3. Alice and Bob apply on the results $h(x)$ and $h(y)$ the $O(\sqrt{n})$ protocol for input of length λ.

In the interactive model we apply on the compressed values the public coins protocol of Rabin and Yao (described in Kushilevitz and Nisan [KN96], see analysis at [BK97]):

Algorithm B.5 Equality Interactive Protocol of Complexity $O(\log \lambda)$

Let $x \in [2^n]$ and $y \in [2^n]$ be Alice's and Bob's inputs.

1. Alice and Bob sample a common CRH $h\colon \{0,1\}^n \to \{0,1\}^\lambda$.
2. Alice chooses a random prime p from the first n^2 primes.
3. Alice sends p and $x \pmod p$ (that is, $O(\log n)$ bits).
4. Bob checks whether $y \pmod p = x \pmod p$ and outputs the answer as a bit.

The communication complexity of the above protocol is $O(\log p) = O(\log n)$ as required.