Compression from Collisions, or why CRHF Combiners have a Long Output

Krzysztof Pietrzak

CWI Amsterdam, The Netherlands

Abstract. A black-box combiner for collision resistant hash functions (CRHF) is a construction which given black-box access to two hash functions is collision resistant if at least one of the components is collision resistant.
In this paper we prove a lower bound on the output length of black-box combiners for CRHFs. The bound we prove is basically tight as it is achieved by a recent construction of Canetti et al [Crypto'07]. The best previously known lower bounds only ruled out a very restricted class of combiners having a very strong security reduction: the reduction was required to output collisions for both underlying candidate hash-functions given a single collision for the combiner (Canetti et al [Crypto'07] building on Boneh and Boyen [Crypto'06] and Pietrzak [Eurocrypt'07]).
Our proof uses a lemma similar to the elegant "reconstruction lemma" of Gennaro and Trevisan [FOCS'00], which states that any function which is not one-way is compressible (and thus uniformly random function must be one-way). In a similar vein we show that a function which is not collision resistant is compressible. We also borrow ideas from recent work by Haitner et al. [FOCS'07], who show that one can prove the reconstruction lemma even relative to some very powerful oracles (in our case this will be an exponential time collision-finding oracle).

1 Introduction

Combiners. A robust black-box $(1,2)$-combiner for some cryptographic primitive α is a construction, which given black-box access to two components, securely implements α if *either* of the two components securely implements α. More generally, for $k \leq \ell$, one can consider black-box (k,ℓ)-combiners which securely implement α, if at least k of the ℓ components the combiner has access to securely implement α. In this introduction, we will mostly talk about $(1,2)$-combiners (and simply call them combiners), but the results in the paper are stated for general (k,ℓ)-combiners.

Combiners for CRHFs. Combiners can be used as a hedge against the failure of a concrete construction: the combiner remains secure as long as at least one of the two combined constructions is not broken. In light of the many recent attacks on popular collision resistant hash functions [20, 21], combiners for CRHFs are of particular practical interest. A function $H : \{0,1\}^* \to \{0,1\}^v$

is collision-resistant, if no efficient algorithm can find two inputs $M \neq M'$ where $H(M) = H(M')$, such a pair (M, M') is called a collision for H.[1]

One trivially gets a $(1, 2)$-combiner for CRHF by simply concatenating the outputs of the components:

$$C^{H_1,H_2}(X) = H_1(X) \| H_2(X). \tag{1}$$

This is a robust combiner for any reasonable definition of "robust", as a collision for $C^{H_1,H_2}(.)$ is also a collision for $H_1(.)$ and $H_2(.)$. Unfortunately the output length ℓ_{out}^C of C^{H_1,H_2} is twice the output length ℓ_{out}^H of its components, which makes this combiner not very useful for practical applications, where the output length is usually a crucial parameter, and doubling it is not an option. The existence of black-box combiners for CRHFs with "short" output length has first been investigated by Boneh and Boyen [2] who showed that no "highly efficient" robust combiner with output length $\ell_{out}^C < 2\ell_{out}^H$ exists. Here "highly efficient" means that the combiner is allowed only one query to each of its components (thus the combiner (1) who achieves $\ell_{out}^C = 2\ell_{out}^H$ is the best "highly efficient" black-box combiner one can hope for). Subsequently, for the more general case where one allows the combiner C any number q_C of oracle gates, a lower bound of $\ell_{out}^C \geq 2\ell_{out}^H - O(\log q_C)$ was proven [16].

In [2, 16] a combiner is defined as a pair (C, P), where the oracle circuit C defines the construction of the combiner, and the oracle PPTM P is the "security proof" for C. The security definition requires that for any hash functions H_1, H_2, and any collision M, M' (i.e. $M \neq M'$ and $C^{H_1,H_2}(M) = C^{H_1,H_2}(M')$), we have that $P^{H_1,H_2}(M, M')$ finds a collision for H_1 and H_2 (here we say that a collision for H_i is found if P makes two queries $X \neq X'$ to H_i where $H_i(X) = H_i(X')$). This is a good definition, as such a (C, P) clearly achieves what one intuitively would require from a robust combiner. But when proving impossibility results, one should try to use a definition which is as general as possible, and ideally should cover (and thus rule out) any black-box construction which would satisfy what one intuitively would consider a robust combiner. In this paper we consider what is arguably the most general definition of black-box combiners for CRHFs.

A General Definition. Informally, we define a randomized black-box combiner for CRHFs as a pair (C, P), where C is a *randomized* oracle circuit, and the oracle PPTM P is the security reduction. For $0 \leq \rho \leq 1$, we say that an oracle $\mathcal{B}$ ρ-breaks C^{H_1,H_2}, if on input some randomness R the oracle $\mathcal{B}$ outputs a collision for $C^{H_1,H_2}(R, .)$ for at least a ρ fraction of the R's. The combiner (C, P) is ρ-robust if for any H_1, H_2 and any $\mathcal{B}$ which ρ-breaks C^{H_1,H_2} the PPTM P in the random experiment $P^{\mathcal{B},H_1,H_2}$ (let us stress that P can query its oracles $\mathcal{B}, H_1, H_2$ *adaptively*) finds a collision for H_1 and a collision for H_2 with high probability.

---

[1] This definition is intentionally kept informal as there are some issues which make it tricky to have a definition for collision-resistant hash-functions which is theoretically and practically satisfying, see e.g. [18] for a discussion.

Thus if (C, P) is ρ-robust, by picking the randomness for C^{H_1,H_2} uniformly at random, with probability at least $1-\rho$ we will get a construction which is secure if either H_1 or H_2 is. A combiner (C, P) is *efficient*, if C and P make a polynomial number of oracle queries, and *robust* if it is ρ-robust with $\rho \in o(1)$.[2]

Remark 1 (on the definition). In practice it's not enough to require that C and P make a polynomial number of queries, one would require that their total running time is polynomial. One would also require ρ-security where ρ is negligible, not only $\rho \in o(1)$. But keep in mind that we want to prove an impossibility result, so using such an "undemanding" definition makes the impossibility result actually stronger.

The Main Result. Theorem 1 in this paper states that no black-box combiners exist whose output length ℓ_{out}^C is significantly smaller than what can be achieved by concatenation. For the special case of $(1, 2)$-combiners, where concatenation achieves a length of $2\ell_{out}^H$, this means that no efficient and robust combiner exists whose output length satisfies $\ell_{out}^C = 2\ell_{out}^H - \omega(\log \ell_{out}^H)$. This result is tight because $\ell_{out}^C = 2\ell_{out}^H - \Theta(\log \ell_{out}^H)$ is achievable as we'll explain in Section 2.

Combining CRHF *Families.* In our definition, the (randomized) combiner is instantiated with two hash-functions H_1, H_2. A seemingly more general definition would allow an instantiation of the combiner with two *families* $\mathcal{H}_1, \mathcal{H}_2$ of hash functions, and only require that the reduction $P^{\mathcal{B},\mathcal{H}_1,\mathcal{H}_2}$ outputs a collision for some $h_1 \in \mathcal{H}_1$ and some $h_2 \in \mathcal{H}_2$. Here the combiner $C^{\mathcal{H}_1,\mathcal{H}_2}(R, M)$ can query different, adaptively chosen functions from $\mathcal{H}_1$ and $\mathcal{H}_2$. Our impossibility also rules out the case where one considers combiners for families as just described, the reason is that we can always view a single hash function $H_b : \{0,1\}^* \to \{0,1\}^v$ as a family $\mathcal{H}_b = \{0,1\}^k \times \{0,1\}^* \to \{0,1\}^v$ (where the first k bits $K \in \{0,1\}^k$ of the input define the hash function $h_b^K \in \mathcal{H}_b$ as $h_b^K(M) = H_b(K\|M)$).

Note that a collision M, M' for any $h_b^K(.) = H_b(K\|.) \in \mathcal{H}_b$ directly gives a collision $K\|M, K\|M'$ for H_b. Thus if (C, P) is not a black-box combiner in our sense, which we prove by showing that there exist $H_1, H_2, \mathcal{B}$ where $P^{\mathcal{B},H_1,H_2}$ does not output collisions for both, H_1 and H_2 (except with negligible probability), it directly follows that $P^{\mathcal{B},\mathcal{H}_1,\mathcal{H}_2}$ will not be able to output collisions for some $h_1^K \in \mathcal{H}_1$ and some $h_2^{K'} \in \mathcal{H}_2$ either.

The Canetti et al Lower Bound. *Randomized* combiners for CRHFs have recently been considered by Canetti et al.[3],[3] who proved a basically tight $\ell_{out}^C \geq 2\ell_{out}^H - O(\log \ell_{out}^H)$ lower bound on the output length for randomized combiners using a definition which is basically equivalent to the one in this paper, but with the restriction, that the reduction P is only allowed a single query to the breaking oracle $\mathcal{B}$. We see no good reason to motivate this restriction on the reduction,

[2] By $\rho \in o(1)$ we mean that $\rho \in o(\ell_{out}^H)/\ell_{out}^H$, i.e. ρ drops below any constant for a sufficiently large security parameter which w.l.o.g. will be the output length ℓ_{out}^H of the components H_i

[3] Let us mention that the main topic of their paper is security amplification, not combiners, for CRHFs.

except for the fact that the existing combiners are of this form. In particular, a reduction which needs, say, any two different collisions[4] for the combiner in order to break the components, would still be perfectly convincing.

Related Work We only consider black-box combiners, and not general combiners, where the combiner gets a full description (e.g. as a circuit) of the underlying primitives and thus is not limited to access the primitives in a black-box manner. This can be justified by noticing that most known cryptographic constructions (not only of combiners) are black-box, which means that the construction only uses the input/output behaviour of the underlying components, and moreover the security proof is black-box, which means that the reduction only uses a successful adversary against the construction in a black-box fashion in order to break the security assumption on the underlying component, such constructions are called "fully black-box" in the taxonomy of [17]. The few exceptions of non black-box constructions (notably the GMW construction of zero-knowledge proofs for NP [6] and Barak's work [1]), are very inefficient. Thus even if non black-box combiners with short output should exist, it would be very surprising if they actually were of any practical relevance. The motivation to restrict oneself to black-box constructions is that it is often feasible to *rule out* such constructions, by using the fact that a black-box reduction is relativizing, i.e. it holds relative to any oracle. Thus a way to rule out the existence of a black-box construction of some primitive α from some primitive β, is to come up with a hypothetical (usually non-efficient) oracle, such that relative to this oracle β exists, but α does not. This technique was first used in the context of cryptographic reductions by Impagliazzo and Rudich, who in their seminal paper [10] prove the impossibility of a black-box construction of key-agreement from one-way functions. Another classical result along this line due to Simon [19] proves the impossibility of constructing CRHFs from one way functions, the breaking oracle in this paper is inspired by this work.

Kim et al. [11] were the first to consider lower bound on the *efficiency* (as opposed to mere feasibility) of black-box constructions. They prove a lower bound on the number of calls to a one-way permutation needed to implement a pseudorandom generator (a thigh bound was subsequently proven in [5]).

The concept of a combiners has first been explicitly considered by Herzberg [9] (who called them "tolerant constructions") and later by Harnik et al. [8], who coined the term "robust combiner". For many natural cryptographic primitives like one-way functions, PRGs or CRHFs $(1, 2)$-combiners are trivially seen to exist. For other primitives like commitments and oblivious transfer the question is open [8, 9, 13, 14].

As mentioned already in the introduction, combiners for CRHFs have been investigated by [2, 3, 16]. Fischlin and Lehmann [4] consider CRHFs combiners in an ideal setting, and in this setting are able to give a construction which

[4] Here "different collision" can either mean two different collisions for $C^{H_1,H2}(R,.)$ and any randomness R, or a collision for $C^{H_1,H_2}(R,.)$ and (a not necessarily different one) for $C^{H_1,H_2}(R',.)$ where $R \neq R'$.

is *more secure* than any of its components. Fischlin, Lehmann and Pietrzak recently constructed a robust $(1,2)$-combiner for hash-functions with output length $2\ell_{out}^{H}$, which simultaneously combines several properties, namely collision-resistance, target collision-resistance, message authentication, pseudorandomness, one-wayness and – at the price of a slightly longer output – indifferentiability from a random oracle [12].

2 Combiners for CRHFs: Definition and Constructions

Notation and some Basic Definitions For $X, Y \in \{0,1\}^*$ we denote with $X\|Y$ the concatenation of X and Y. For $a \in \mathbb{N}$ we denote with $[a]$ the set $\{0,1,\ldots,a-1\}$ and $\langle a\rangle_b$ denotes the binary representation of a, padded with 0's to length b, e.g. $\langle 5\rangle_6 = 000101$. A pair M, M' is a *collision* for a function F if $F(M) = F(M')$ and $M \neq M'$. We call M, M' a *pseudocollision* for F if $F(M) = F(M')$ (but not necessarily $M \neq M'$). With $X \stackrel{*}{\leftarrow} \mathcal{X}$ we denote that X is assigned a value chosen uniformly at random from the set $\mathcal{X}$.

PPTM stands for Probabilistic Polynomial time Turing Machine. An *oracle* PPTM A (oPPTM for short) is a PPTM with an extra oracle tape, where $A^{\mathcal{O}_1,\ldots,\mathcal{O}_z}$ denotes the random experiment where A runs having access to the oracles $\mathcal{O}_1,\ldots,\mathcal{O}_z$ via its oracle tape: A can write a query (i, X) on the tape, and in the next step the value $\mathcal{O}_i(X)$ is written on the tape. Let

$$\mathsf{qry}^{\mathcal{O}_i}(A^{\mathcal{O}_1,\ldots,O_z})$$

denote the queries that A makes to the oracle $\mathcal{O}_i$. In this paper, the oracles will always be hash functions $H_1, H_2, \ldots$ and possibly a breaking oracle $\mathcal{B}$. The collision predicate col^{H_i} is defined for the random experiment $A^{H_1,\ldots,H_\ell}$ and holds if A finds a collision for H_i, i.e. A makes two distinct queries X, X' to H_i where $H_i(X) = H_i(X')$, formally[5]

$$\mathsf{col}^{H_i}(A^{H_1,\ldots,H_z}) \iff \exists X, X' \in \mathsf{qry}^{H_i}(A^{H_1,\ldots,H_z}) : X \neq X' \wedge H_i(X) = H_i(X')$$

More generally, for $\mathcal{H} \subseteq \{H_1,\ldots,H_\ell\}$ we define the predicate

$$\mathsf{col}^{\mathcal{H}}(A^{H_1,\ldots,H_z}) \iff \forall H_i \in \mathcal{H} : \mathsf{col}^{H_i}(A^{H_1,\ldots,H_z})$$

which holds if A finds a collisions for all H_i in $\mathcal{H}$. Finally

$$\mathsf{col}_t(A^{H_1,\ldots,H_z}) \iff \exists \mathcal{H} \subseteq \{H_1,\ldots,H_\ell\}, |\mathcal{H}| = t : \mathsf{col}^{\mathcal{H}}(A^{H_1,\ldots,H_z})$$

holds if A finds collisions for at least t of the H_i's.

Definition 1 (Randomized Black-Box Combiner For CRHFs). CONSTRUCTION: *A randomized (k,ℓ)-combiner for CRHFs is a pair (C, P) where C is an oracle circuit $C : \mathcal{R} \times \{0,1\}^m \to \{0,1\}^n$ and P is an oracle PPTM.*

[5] Note that we e.g. write simply $\mathsf{col}^H(A^H)$ to denote "the predicate col^H is satisfied in the random experiment A^H", with $\neg\mathsf{col}^H(A^H)$ we denote the complementary event.

REDUCTION: *An oracle $\mathcal{B}$ ρ-breaks $C^{H_1,\dots,H_\ell}$ if $\mathcal{B}(R)$ outputs a collision for $C^{H_1,\dots,H_\ell}(R,.)$ for at least a ρ-fraction of the possible choices of the randomness $R \in \mathcal{R}$, and $\perp$ on the remaining inputs.*

(C,P) is ρ-robust (where ρ can be a function of $v \in \mathbb{N}$) if for all $H_1,\dots,H_\ell : \{0,1\}^* \to \{0,1\}^v$ and any oracle $\mathcal{B}$ which ρ-breaks $C^{H_1,\dots,H_\ell}$ the PPTM P in the random experiment $P^{\mathcal{B},H_1,\dots,H_\ell}$ finds collisions for at least $\ell-k+1$ of the H_i's with probability at least .9, i.e.

$$\Pr_{H_1,\dots,H_\ell,P's\ coins}[\mathsf{col}_{\ell-k+1}(P^{\mathcal{B},H_1,\dots,H_\ell})] \geq .9 \tag{2}$$

EFFICIENCY: *Let q_C denote the number of oracle gates in C, and q_P be an upper bound on the number of oracle queries made by P, where we do not count oracle queries to $\mathcal{B}$ where the answer is $\perp$ (as we want to prove a negative result, not accounting for such queries makes the negative result stronger). Then the combiner (C,P) is efficient if q_C and q_P are polynomial in v.*

SECURITY: *An efficient combiner (C,P) is robust, if it is ρ-robust where $\rho = \rho(v)$ is smaller than any positive constant for sufficiently large v.*

Remark 2 (on the constant .9). The probability .9 in (2) is over the random coins of P. We chose to fix this probability to the arbitrary constant .9 instead of adding an additional parameter in the security definition, as the constant .9 can be replaced with any value ϵ where ϵ is noticeable[6] and bounded away from 1 by some exponentially small amount, by changing the running time of P only by a polynomial factor. The reason is that if some efficient combiner (C,P) satisfies (2) for some ϵ (instead .9), then for any $z = poly(v)$, we get an efficient combiner (C,P_z) which satisfies (2) with probability $1-(1-\epsilon)^z$, where P_z simply simulates P z times using fresh random coins for each run.

Concatenation Combiner. We trivially get a robust and very efficient (k,ℓ)-combiner, by concatenating the output of any $\ell-k+1$ of the components.

$$C^{H_1,\dots,H_\ell}(R,M) = H_1(M)\|H_2(M)\|\dots\|H_{\ell-k+1}(M). \tag{3}$$

This combiner is an ρ-robust (k,ℓ)-combiner for any $\rho > 0$, where

$$n = (\ell-k+1)v \qquad q_C = \ell-k+1 \qquad q_P = 1$$

The reduction P achieving the above parameters, simply queries the oracle $\mathcal{B}$ on distinct $R \in \mathcal{R}$ until it gets a collision (as $\rho > 0$, there will be at least one).

Random Concatenation Combiner As a generalization of the previous combiner, we can consider the combiner $C : [\binom{\ell}{c}] \times \{0,1\}^m \to \{0,1\}^n$ where we concatenate the output of c randomly chosen components. For $c < \ell-k+1$ this combiner has shorter output than (3), but also is only ρ-robust for a ρ which is bounded away from 0 by a constant, and thus is not a "robust combiner". The only reason we mention this construction here is to make clear, that the upper bound on ρ which we will need in our impossibility result is necessary.

[6] i.e. at least $1/poly(v)$ for some positive polynomial poly.

Below each R in the randomness space $[\binom{\ell}{c}]$ is parsed as a c element subset $1 \leq R_1 < R_2 < \ldots < R_c \leq \ell$ of $[\ell]$.

$$C^{H_1,\ldots,H_\ell}(R,M) = H_{R_1}(M)\|H_{R_2}(M)\|\ldots\|H_{R_c}(M)$$

In the full version of the paper we prove that this combiner is a ρ-robust (k,ℓ)-combiner for any $\rho > \binom{\ell-k}{c}/\binom{\ell}{c}$ with parameters

$$n = cv \qquad q_C = c \qquad q_P = \ell - k + 2 - c$$

Thus efficient ρ-robust (k,ℓ)-combiners with output length $(\ell-k+1)v$ exists for any $\rho > 0$, on the other extreme, we can get (by setting $c = 1$ in the above construction) ρ-robust combiners for any $\rho > 1 - k/\ell$ with an output length of only v. This can be slightly improved as we'll describe now.

The Canetti et al $(1,1)$-Combiner. A remarkable construction of Canetti et al [3] is a $(1,1)$ black-box Combiner S which, from any CRHF H with range v, constructs a CRHF S^H with range $v - \Delta$. Unfortunately, for *efficient* combiners, Δ must be logarithmic, as the running time of S increases exponentially in Δ.

We will shortly sketch the idea of the Canetti et al. combiner, for the detailed construction of that combiner we refer the reader to the original paper [3]. Let $H : \{0,1\}^w \to \{0,1\}^v$ be a hash function. First one finds a string $\gamma \in \{0,1\}^\Delta$ where for a random z, the prefix of $H(z)$ is γ with good probability.[7] Let $\widetilde{H}(z)$ denote $H(z)$ but with the first Δ bits deleted, and let

$$Z := \{z \in \{0,1\}^w : \text{ the prefix of } H(z) \text{ is } \gamma\}$$

Note that any collision z, z' for $\tilde{H}$ where $z, z' \in Z$ is also a collision for H, as

$$H(z) = \gamma\|\widetilde{H}(z) = \gamma\|\widetilde{H}(z') = H(z')$$

Thus we have constructed a CRHF $\widetilde{H} : Z \to \{0,1\}^{v-\Delta}$ from a CRHF $H : \{0,1\}^w \to \{0,1\}^v$. This is almost a $(1,1)$-combiner with output length $v - \Delta$, except that the domain is some strange set Z. We must somehow map $\{0,1\}^{w'}$ where $w' > v$ injectively to a (subset) of Z in order to get a CRHF $\{0,1\}^{w'} \to \{0,1\}^v$. As shown in [3] this can be achieved, albeit inefficiently in time 2^Δ.

One can replace the H_i's with S^{H_i} in the combiners considered before in order to get shorter output, e.g. for the concatenation combiner (3) we get

"Shrinked" Concatenation Combiner: The combiner (with S as above)

$$C^{H_1,\ldots,H_\ell}(R,M) = S^{H_1}(M)\|S^{H_2}(M)\|\ldots\|S^{H_{\ell-k+1}}(M) \tag{4}$$

satisfies for any $\rho > 0$

$$n = (\ell-k+1)(v-\Delta) \qquad q_C = 2^{O(\Delta)}(\ell-k+1) \qquad q_P = O(2^\Delta)$$

[7] The expected probability for a random γ is $2^{-\Delta}$, we're fine with anything not much smaller than that, say $2^{-\Delta-1}$, such a good γ can be found by sampling.

Main Theorem. In this paper we'll prove that the bound achieved by the combiners (4) is basically tight.

Theorem 1 (Main). *If (C, P), where*

$$C : \{0,1\}^m \to \{0,1\}^n$$

is an efficient and robust randomized (k,ℓ)-combiner for CRHFs with range $\{0,1\}^v$, then

$$n \geq (\ell - k + 1)v - O(\log(q_C)).$$

This theorem is stated in asymptotic terms so it is easy to parse, but we prove a quantitative statement. The quantitative statements are given by Proposition 3 for the special case of $(1,1)$-combiners, and in Proposition 4 for general (k,ℓ)-combiner. In particular, the exact meaning of "efficient" in the theorem above is given by equation (30), where $q_P^{\mathcal{B}}$ and q_P^H denote an upper bound on the number of oracle queries the reduction P makes to the breaking oracle and to the candidate hash functions respectively, so $q_P = q_P^{\mathcal{B}} + q_P^H$. Throughout the paper we assume w.l.o.g. that $q_P^{\mathcal{B}}, q_P^H$ and q_C are at least one.

Lower Bounds for Restricted Combiners. A result analogous to the statement of Theorem 1 has been proven for restricted cases of combiners. Starting with [2], who proved it for *deterministic* combiners (i.e. where $\mathcal{R}$ in Definition 1 is empty), and where the construction C was only allowed to query each H_i exactly once. A simpler proof without the latter restriction (but still deterministic) was subsequently given in [16]. The proof was further simplified in [3], who also for the first time considered the randomized case, but under the restriction that the reduction P queries the breaking oracle at most once. This special case seems much easier to prove than the general one. As the main idea behind the proof of the special case, which is a probabilistic argument, is also used in the proof of the general case, we give the full statement and proof of the special case below.

Proposition 1 (following [3]). *For some n, m, v with $m > n$, assume that (C, P) where*

$$C : \{0,1\}^m \to \{0,1\}^n$$

is a 1-robust (k,ℓ)-combiner for CRHFs with range $\{0,1\}^v$, with the additional constraint that P is querying the breaking oracle only once. Let ϵ denote the success probability (over P's random coins) of P, i.e. for any breaking oracle $\mathcal{B}$ which on input R outputs a collision for $C^{H_1,\ldots,H_\ell}(R,.)$ [8]

$$\forall H_1, \ldots, H_\ell : \Pr_{P's\ coins}[\mathsf{col}_{k+1}(P^{\mathcal{B},H_1,\ldots,H_\ell})] \geq \epsilon$$

[8] Here Remark 2 (after Def.1) does not apply, as now we can't run P several times to amplify ϵ as we're only allowed one query to $\mathcal{B}$. So unlike in the general case where we arbitrarily set $\epsilon = .9$, here it is necessary to keep ϵ as a parameter.

Then the output length n of C satisfies

$$n \geq (\ell - k + 1)(v + 1 - 2\log q_P) - \log(\binom{\ell}{\ell - k + 1}) + \log(\epsilon) + 1 \tag{5}$$

Before we prove the proposition, let us remark that for the practically relevant case where P is efficient and ϵ is noticeable, (5) can be written as

$$n \geq (\ell - k + 1)(v - O(\log v))$$

which, up to the constant hidden in the O term, matches parameters of the combiner (4).

Proof. We will only prove the case for $k = 1$ and $\ell = 2$ and explain at the end how to adapt the proof for the general k and ℓ.

Let A be any oracle PPTM making at most q_A oracle queries and $H : \{0,1\}^* \rightarrow \{0,1\}^v$ be uniformly random. The probability that any two (distinct) queries made by A to H give a collision for H is $1/2^v$, taking the union bound over all $q_A(q_A - 1)/2$ possible pairs of queries

$$\Pr_{H,A's\ coins}[\mathsf{col}^H(A^H)] \leq q_A(q_A - 1)/2^{v+1} < q_A^2/2^{v+1}. \tag{6}$$

Now consider an oracle PPTM A which expects two oracles, making at most q_A queries to each of them. Let $H_1, H_2 : \{0,1\}^* \rightarrow \{0,1\}^v$ be uniformly random and independent. As the H_i's are independent, the probability that P will find collisions for both is the product of what we had in eq.(6).

$$\Pr_{H_1,H_2,A's\ coins}[\mathsf{col}^{H_1,H_2}(A^{H_1,H_2})] \leq (q_A^2/2^{v+1})^2 \tag{7}$$

Now let (C, P) be a combiner as in the statement of the proposition. Let A be an oracle PPTM where A^{H_1,H_2} simulates $P^{\mathcal{B},H_1,H_2}$, but answers the (single) $\mathcal{B}$ query R made by P with random $M \xleftarrow{*} \{0,1\}^m, M' \xleftarrow{*} \{0,1\}^m$. Note that P will output collisions for H_1, H_2 with probability ϵ conditioned on the event that M, M' is a collision for $C^{H_1,H_2}(R, .)$.

$$\begin{aligned}
&\Pr_{H_1,H_2,A's\ coins}[\mathsf{col}^{H_1,H_2}(A^{H_1,H_2})] \\
&\geq \Pr[\mathsf{col}^{H_1,H_2}(P^{\mathcal{B},H_1,H_2})] \cdot \Pr[M \neq M' \wedge C^{H_1,H_2}(R, M) = C^{H_1,H_2}(R, M')] \\
&\geq \epsilon \cdot (2^{-n} - 2^{-m}) \geq \epsilon \cdot 2^{-n+1}
\end{aligned} \tag{8}$$

Where in the last step we used $m > n$ which holds as C is shrinking. Now by (7) and (8) we must have $\epsilon \cdot 2^{-n+1} \leq (q_P^2/2^{v+1})^2$, solving for n gives

$$n \geq 2(v + 1 - 2\log q_P) + \log(\epsilon) + 1$$

which is (5) for the case where $k = 1, \ell = 2$. For the general case of (k, ℓ)-combiners, we can similarly upper and lower bound the probability of a PPTM A in finding collision for at least $\ell - k + 1$ of its ℓ oracles as

$$\epsilon \cdot 2^{-n+1} \leq \Pr_{H_1,\ldots,H_\ell,A's\ coins}[\mathsf{col}_{\ell-k+1}(A^{H_1,\ldots,H_\ell})] \leq \binom{\ell}{\ell - k + 1}(q_P^2/2^{v+1})^{\ell-k+1}$$

Solving this inequality for n then gives

$$n - 1 \geq (\ell - k + 1)(v + 1 - 2\log q_P) - \log(\binom{\ell}{\ell - k + 1}) + \log(\epsilon). \qquad \square$$

3 Proof Outline

We will prove our main result gradually, introducing new techniques and ideas in each step. First, in Lemma 1 we show that a uniformly random function is collision resistant, using the fact that such a function cannot be compressed. Based on this technique, we then prove Proposition 3 which implies Theorem 1 for the special case $k = \ell = 1$. Finally, Proposition 4 proves the general case. Due to space reasons, the proof of Proposition 4 is only given in the full version of the paper [15].

Collisions imply Compressibility, Section 4. Gennaro and Trevisan [5] give a very elegant proof that a uniformly random permutation $\pi : \{0,1\}^v \to \{0,1\}^v$ is one-way against poly-size, non-uniform adversaries. On a high level, they show that if P is an efficient[9] adversary which inverts π on many inputs, i.e. for many x we have $A^\pi(\pi(x)) = x$, then π has a "short" description relative to P. This is impossible as a uniformly random π is incompressible, and thus such an P cannot exist (i.e. π is one-way).

We adapt this proof in order to show that a uniformly random *function* $H : \{0,1\}^w \to \{0,1\}^v$ is *collision resistant*. This has been independently discovered by the authors of [7], the proof given in this paper is due to Thomas Holenstein (via personal communication with Iftach Haitner), and is much simpler than the one we had originally.

Lower Bounds for Black-Box Combiners via Incompressibility. The just sketched proof is by no means the easiest way to show that a uniformly random function is collisions resistant.[10]

The advantage of such a "incompressibility based" proof is that it extends to the case where P additionally gets access to a carefully defined "combiner breaking" oracle $\mathcal{B}$, which itself can make much more queries to the hash function(s) than what is needed to find collisions for uniformly random functions with output length v bits (which means roughly $2^{v/2}$ queries), as we'll explain below. This approach is inspired by a recent work of Haitner et al [7], the Gennaro-Trevisan reconstruction lemma [5] and Simon's breaking oracle [19].

[9] Here efficient means that the number of oracle queries made by P must be much smaller than what would be required to invert π by brute force search (but can still be exponential).

[10] The straight forward way to prove this, is to argue that for any two distinct queries X_a, X_b made by P we have $\mathsf{Pr}[H(X_a) = H(X_b)] = 2^{-v}$, and thus by taking the union bound over all $q(q-1)/2$ pairs of queries, the probability that there exist any X_a, X_b where $H(X_a) = H(X_b)$ is at most $q(q-1)/2^{v+1}$.

Lower bound for $(1,1)$-combiners, Section 5. In order to rule out the existence of an efficient ρ-robust black-box combiner (C,P) with output length $n = v - \omega(\log v)$, one must come up with oracles $H, \mathcal{B}$ such that

- $C^H : \{0,1\}^r \times \{0,1\}^m \rightarrow \{0,1\}^n$ is not collision resistant, in the sense that $\mathcal{B}(R)$ outputs a collision for $C^H(R,.)$ on at least a ρ-fraction of the $R \in \{0,1\}^r$.
- $H : \{0,1\}^w \rightarrow \{0,1\}^v$ is collision resistant (even relative to $\mathcal{B}$), in the sense that the probability that $P^{H,\mathcal{B}}$ finds a collision (where the probability is over the random coins of P) is small, which means < 0.9 (cf. Remark 2).

The oracle hash function $H : \{0,1\}^w \rightarrow \{0,1\}^v$ is chosen uniformly at random. The breaking oracle $\mathcal{B}$ samples, for each possible input $R \in \{0,1\}^r$, a random pseudocollision Z_R, Z'_R for $C^H(R,.)$. On input R the oracle $\mathcal{B}$ outputs Z_R, Z'_R if this is a "safe" collision, by which we mean that the H queries needed in order to evaluate $C^H(R, Z_R)$ and $C^H(R, Z'_R)$ do not contain a collision for H. If the collision is not safe, then $\mathcal{B}(R)$ outputs $\perp$.

Using the fact that the output length n of C^H is by $\omega(\log v)$ bits shorter than the output length of H, one can show (using a probabilistic argument like in the proof of Proposition 1), that with high probability most collisions Z_R, Z'_R will be safe, and thus $\mathcal{B}$ will ρ-break C^H for a ρ which is exponentially close to 1. This is the only part of the proof where we use the fact that C has short output.

It remains to prove that P cannot find collisions for H, even with the powerful combiner breaking oracle $\mathcal{B}$. Intuitively, $\mathcal{B}$ should not be of much help in finding collision for H, as it only returns random collisions for $C^H(R,.)$ which are "safe" (as described above), and thus do not (at least trivially) give collisions for H. To actually prove this, we show that if $P^{H,\mathcal{B}}$ finds collisions with high probability, then we can use P to compress H, which is impossible as H is uniformly random, thus such a P cannot exist.

Lower bound for (k,ℓ)-combiners, Section 5. To rule out the existence of an efficient ρ-robust (k,ℓ)-black-box combiner (C,P) with output length $n = (\ell - k + 1)v - \omega(\log v)$, we will construct ℓ hash functions $H_1, \ldots, H_\ell \stackrel{\text{def}}{=} H^\ell$ and a breaking oracle $\mathcal{B}$ which ρ-breaks C^{H^ℓ}, but at least k of the H_i's are collision resistant even relative to $\mathcal{B}$. The ρ we achieve will be exponentially close to $1/\binom{\ell}{k}$, which is tight because (as explained in the last section) for $\rho > 1/\binom{\ell}{k}$ combiners with output length only $(\ell - k + 1)v$ exist. The $H^\ell = H_1, \ldots, H_\ell : \{0,1\}^w \rightarrow \{0,1\}^v$ are chosen uniformly at random. The breaking oracle $\mathcal{B}$ samples, for each $R \in \{0,1\}^r$ a collision Z_R, Z'_R for $C^{H^\ell}(R,.)$ (or, a pseudocollision to be precise, as there's a tiny 2^{-m} probability that $Z_R = Z'_R$). We say that Z_R, Z'_R is a safe collision for H_i, if the evaluation of $C^{H^\ell}(R,.)$ on inputs Z_R, Z'_R does not contain a collision of H_i. By a probabilistic argument, one can show that with high probability a random collision will be safe for at least k of the H_i's (here we need the fact that the output length of C is short). This again implies that there exists a subset $\Gamma \subset \{1, \ldots, \ell\}$ of size k, such that for (almost) a $1/\binom{\ell}{k}$ fraction of the R's, let's call it $\mathcal{R}_\Gamma$, the collision Z_R, Z'_R is safe for all the H_i with $i \in \Gamma$.

Now $\mathcal{B}$ on input R outputs Z_R, Z'_R if $R \in \mathcal{R}_\Gamma$, and $\perp$ otherwise. Intuitively, the H_i where $i \in \Gamma$ should be still be collision resistant even relative to $\mathcal{B}$. To prove this we show that if an efficient P exists where $P^{\mathcal{B},H^\ell}$ finds a collision for any H_i where $i \in \Gamma$ with high probability, then this H_i can be compressed, which is impossible as H_i is uniformly random, and thus such a P cannot exist.

4 Collisions imply Compressibility

For a function $H : \{0,1\}^w \to \{0,1\}^v$, we denote with $\widetilde{H} \in \{0,1\}^{2^w v}$ the function table of H, which is a binary $2^w \times v$ matrix. We number the rows from 0 to $2^w - 1$, thus the i'th row contains the value $H(i)$. Such a function table can be uniquely encoded by a bit-string of length $2^w v$.

A random variable H can be compressed to s bits, if there exists a pair $\mathsf{com}, \mathsf{dec}$ of functions (possibly probabilistic using joint randomness) such that for any $t \in \mathbb{N}$ and $\widetilde{H}_1, \ldots, \widetilde{H}_t$ being independent instantiations of H, we have

$$\mathop{\mathsf{E}}_{\widetilde{H}_1,\ldots,\widetilde{H}_t}[|\mathsf{com}(\widetilde{H}_1, \ldots, \widetilde{H}_t)|] \leq t \cdot s \tag{9}$$

$$\Pr_{\widetilde{H}_1,\ldots,\widetilde{H}_t}[\mathsf{dec}(\mathsf{com}(\widetilde{H}_1, \ldots, \widetilde{H}_t)) = \widetilde{H}_1, \ldots, \widetilde{H}_t] = 1 \tag{10}$$

As already proved by Shannon, a function table which is chosen uniformly at random, cannot be compressed, i.e.

Proposition 2. *A uniformly random function $H : \{0,1\}^w \to \{0,1\}^v$ cannot be compressed to less than $2^w v$ bits.*

By the following proposition, any function H for which there exists an efficient collision finding algorithm P, can be compressed.

Lemma 1. *Let P be an oracle PPTM which makes at most q_P oracle queries. Let H be a random variable taking as value functions $\{0,1\}^w \to \{0,1\}^v$. For $0 \leq \delta \leq 1$, if P finds a collision with probability δ:*

$$\Pr_{H,P's\ coins}[\mathsf{col}^H(P^H)] = \delta \tag{11}$$

then H can be compressed to

$$1 + 2^w v - \delta(v - 2\log(q_P)) \qquad \textit{bits.} \tag{12}$$

Using Proposition 2 we get the following Corollary

Corollary 1. *Let $H : \{0,1\}^w \to \{0,1\}^v$ be uniformly random, then any P which for some $\delta > 0$ satisfies eq. (11) makes at least $q_P \geq 2^{v/2-1/2\delta}$ oracle queries.*

Proof (of Corollary). If H is uniformly random, then by Proposition 2 expression (12) is at least $2^w v$ which means $1 \geq \delta(v - 2\log(q_P))$, or equivalently $v/2 - 1/2\delta \leq \log q_P$ which implies $q_P \geq 2^{v/2-1/2\delta}$ by exponentiating on both sides. $\square$

Proof (of Lemma 1). Consider a variable H taking as values functions $\{0,1\}^w \to \{0,1\}^v$ and any PPTM P making at most q_P oracle queries. If P^H does not find a collision for H, then we do not compress at all, in this case $\mathsf{col}(\widetilde{H})$ is simply a 0 followed by $\widetilde{H}$. Otherwise let $X_1, X_2, \ldots$ denote the oracle queries made by P^H and let X_{c_1}, X_{c_2} where $c_1 < c_2$ denote the collision found. Let $\widetilde{H}^- \in \{0,1\}^{(2^w - c_2)v}$ denote $\widetilde{H}$ with the rows $X_1, \ldots, X_{c_2}$ (containing the value $H(X_1), \ldots, H(X_{c_2})$) deleted. Now $\mathsf{com}(\widetilde{H})$ is a 1 followed by an encoding of the indices c_1, c_2 followed by the first $c_2 - 1$ oracle answers $H(X_1), \ldots, H(X_{c_2-1})$ and finally $\widetilde{H}^-$, i.e.

$$\mathsf{com}(\widetilde{H}) = \begin{cases} 0\|\widetilde{H} & \texttt{if } \neg\mathsf{col}(P^H) \\ 1\|\langle c_1\rangle_{\log_{q_P}}\|\langle c_2\rangle_{\log_{q_P}}\|H(X_1)\|\ldots\|H(X_{c_2-1})\|\widetilde{H}^- & \texttt{if } \mathsf{col}(P^H) \end{cases}$$

On input more than one function table, com simply compresses each function table separately, and then concatenates the outputs, i.e.

$$\mathsf{com}(\widetilde{H}_1, \ldots, \widetilde{H}_t) = \mathsf{com}(\widetilde{H}_1)\|\ldots\|\mathsf{com}(\widetilde{H}_t)$$

Before we describe the decompression algorithm, let us check that this compression really achieves the length as claimed in eq.(12). The output length of $\mathsf{com}(\widetilde{H})$ is $1+2^w v$ if P does not find a collision for H, which by assumption happens with probability $1-\delta$. Otherwise the length is $1 + (c_2 - 1)v + (2^w v - c_2)v + 2\log q_P$, which gives an expected length of

$$\mathsf{E}[|\mathsf{com}\widetilde{H}|] = 1 + (1-\delta)2^w v + \delta\left((2^w - 1)v + 2\log q_P\right) = 1 + 2^w v - \delta(v - 2\log q_P)$$

as claimed. The decompression algorithm dec, on input $T = \mathsf{com}(\widetilde{H}_1, \ldots, \widetilde{H}_t)$ first parses T into $\mathsf{com}(H_1)$ to $\mathsf{com}(H_t)$ which can be done as the length (there are only 2 possibilities) of $\mathsf{com}(H_1)$ can be uniquely determined reading only the first bit. We can then strip off $\mathsf{com}(H_1)$, the first bit of the remaining string determines the length of $\mathsf{com}(H_2)$, and so on. We thus must only show how to decompress a single compressed function table $T = \mathsf{com}(\widetilde{H})$. On input $T = \mathsf{com}(\widetilde{H})$, dec parses T as $b\|T'$, where $b \in \{0,1\}$. If $b = 0$ the output is T' and we are done. Otherwise parse T' as

$$\langle c_1\rangle_{\log_{q_P}}\|\langle c_2\rangle_{\log_{q_P}}\|H(X_1)\|\ldots\|H(X_{c_2-1})\|\widetilde{H}^-$$

Now simulate P^H up to the point where P asks the c_2'th oracle query X_{c_2}.[11] Note that we can answer the first $c_2 - 1$ oracle queries made by P as we know $H(X_1), \ldots, H(X_{c_2-1})$. Now, by construction we also know $H(X_{c_2})$, as it is equal to $H(X_{c_1})$. We can now reconstruct (and output) $\widetilde{H}$ from the reduced table $\widetilde{H}^-$ as we know all missing values $H(X_1)$ to $H(X_{c_2})$ and also the positions X_1 to X_{c_2} where to insert them in $\widetilde{H}^-$ in order to get $\widetilde{H}$.

[11] As P can be probabilistic, we need com and dec to use the same random coins for P. Alternatively, we can just fix the randomness of P as to maximize $\mathsf{Pr}[\mathsf{col}^H(P^H)]$.

Before we continue proving Theorem 1, we need a few more definitions.

Definition 2 (safe collisions, the predicate $\mathsf{safeCol}$). *Let $H^\ell = H_1, \dots, H_\ell$ be ℓ hash functions and A be an oPPTM. We say that Z, Z' is a safe collision for H_i (with respect to A^{H^ℓ})*

1. $A^{H^\ell}(Z) = A^{H^\ell}(Z')$ *(but not necessarily $Z \neq Z'$)*
2. *during the evaluation of $A^{H^\ell}(.)$ on inputs Z and Z', there are no two queries $X \neq X'$ to H_i where $H_i(X) = H_i(X')$.*

We have $\mathsf{safeCol}_{H_i}^{A^{H^\ell}}(Z, Z')$ if Z, Z' is a safe collision. For any $1 \leq k \leq \ell$, $\mathsf{safeCol}_{k}^{A^{H^\ell}}(Z, Z')$ holds if for at least k different i's, $\mathsf{safeCol}_{H_i}^{A^{H^\ell}}(Z, Z')$ holds.

Intuitively, when given $Z \neq Z'$ where $\mathsf{safeCol}_{H_i}^{A^{H^\ell}}(Z, Z')$, one learns a collision for A^{H^ℓ}, but this collision does not (at least trivially) give us a collision for H_i.

Definition 3 ($\prec$). *If we consider a random experiment where some oPPTM runs making queries to its oracle(s). Then for two queries X, Y (not necessarily to the same oracle) we denote by $X \prec Y$ that the query X is made before the query Y is made.*

5 Lower bounds

Lower bound for $(1,1)$-combiners. In this section we prove a Proposition which implies Theorem 1 for the special case $k = \ell = 1$. The word "combiner" is a bit misleading in this case, "shrinker" would be more appropriate, as we ask for a construction which given access to a hash function H with range $\{0,1\}^v$, gives a hash function whose output length n is "significantly" shorter than v.

Proposition 3 (implies Thm.1 for the special case $k = \ell = 1$). *Let $C : \{0,1\}^r \times \{0,1\}^m \to \{0,1\}^n$ be an oracle circuit with input range $m := v + 1$ bits and with q_C oracle gates, where for some $t > 0$*

$$n := v - 2\log(q_C) - t \tag{13}$$

then, if for some oracle PPTM P (which makes $q_P^{\mathcal{B}}$ oracle calls to the breaking oracle and q_P^H oracle calls to the components) it is the case that (C, P) is a ρ-robust $(1,1)$-combiner with $\rho := 1 - 2^{-t+3}$, then for some constant $\alpha > 0$

$$v \leq \log q_P^{\mathcal{B}} + \log q_C + 2(\log(q_P^H + \alpha q_C q_P^{\mathcal{B}})) + 6 \tag{14}$$

or equivalently,

$$2^v \leq q_P^{\mathcal{B}} \cdot q_C \cdot (q_P^H \cdot \alpha q_C q_P^{\mathcal{B}})^2 \cdot 64$$

in particular, (C, P) is not efficient, as by the above, either C or P must make an exponential number of queries.

Remark 3 (on the constant α). A concrete bound on α in (14) can be determined from the proof of Lemma 4 given in the full version of the paper. A rough estimate suggests that setting $\alpha = 1000$ is far on the safe side. This seems large, but note that only the logarithm of α appears in the expression.

Remark 4 (on the input length). Proposition 3 only rules out combiners which hash their input down by $m - n = t + 2\log q_P + 1$ bits. This implies impossibility for the general case, where the input length can be arbitrary as long as the combiner is shrinking. The reason is that using the Merkle-Damgård construction, one can get a CRHF with any input length from a CRHF which hashes down only one bit.

The Oracle. We now define the oracles, which consist of the hash function H and the breaking oracle $\mathcal{B}$. The oracle H is sampled uniformly at random from all functions $\{0,1\}^w \to \{0,1\}^v$. The oracle $\mathcal{B}$ will be completely defined by a function $\phi : \{0,1\}^* \to \{0,1\}^m$ which we sample uniformly at random. This ϕ defines for each randomness $R \in \{0,1\}^r$ a pseudocollision[12] Z_R, Z'_R for $C^H(R,.)$ as follows: $Z_R := \phi(R)$ and $Z'_R := \phi(R\|\langle i\rangle)$, where i is the smallest integer such that $C^H(R, Z_R) = C^H(R, Z'_R)$. The input/output behavior of oracle $\mathcal{B}$ is now defined as

$$\mathcal{B}(R) = \begin{cases} Z_R, Z'_R & \texttt{if } \mathsf{safeCol}_H^{C^H(R,.)}(Z_R, Z'_R) \\ \perp & \texttt{otherwise} \end{cases}$$

So $\mathcal{B}(R)$ outputs Z_R, Z'_R only if this is a safe collision.

To prove that $\mathcal{B}$ breaks the security of any combiner, we'll need the following technical lemma (for the special case $\ell = 1$), which states that a randomly sampled collision for a combiner C^{H^ℓ} will be safe for many of the $H_\ell = H_1, \ldots, H_\ell$. For how many exactly of course depends on the output length of C. For space reasons we only prove this lemma in the full version.

Lemma 2. *For any oracle circuit $C : \{0,1\}^m \to \{0,1\}^n$ with q_C oracle gates, and ℓ independent uniformly random functions $H^\ell = H_1, \ldots, H_\ell : \{0,1\}^* \to \{0,1\}^v$. For X, X', sampled as $X \xleftarrow{*} \{0,1\}^m$ and $X' \xleftarrow{*} C^{H^\ell}(X)^{-1}$, then for $k \leq \ell$*

$$\mathsf{Pr}[\mathsf{safeCol}_k^{C^{H^\ell}}(X, X')] \geq 1 - 2^{n-m} - (q_C(q_C-1))^{\ell-k+1} \cdot \binom{\ell}{\ell-k+1} 2^{n-(\ell-k+1)\cdot v}$$

$\mathcal{B}$ $1 - 2^{-t+3}$ breaks C^H. Let $I_R = 1$ if $\mathcal{B}(R) \neq \perp$ and $I_R = 0$ otherwise. From Lemma 2 (for $\ell = 1$) if follows that (recall that ϕ is the randomness used by $\mathcal{B}$)

$$\mathop{\mathsf{Pr}}_{H,\phi}[I_R = 0] \leq 2^{n-v-1} + q_C(q_C - 1) \cdot 2^{n-v+1} < q_C^2 \cdot 2^{n-v+1} \tag{15}$$

Note that $\mathcal{B}$ ρ-breaks C^H, where ρ is the fraction of R's for which $\mathcal{B}(R) \neq \perp$. By (15) ρ is a random variable with expectation

$$\mathsf{E}_{H,\phi}[\rho] = 2^{-r} \sum_{R\in\{0,1\}^r} \mathop{\mathsf{Pr}}_{H,\phi}[I_R = 1] > 1 - q_C^2 \cdot 2^{n-v+1} = 1 - 2^{-t+1}$$

---

[12] Recall that X, X' is a pseudocollision for F if $F(X) = F(X')$ but (unlike for collisions) we must not necessarily have $X \neq X'$.

where in the last step we used (13). Applying the Markov inequality,[13] we get $\mathsf{Pr}[\rho < 1 - \gamma 2^{-t+1}] \le 1/\gamma$ for any $\gamma > 0$, we will use this bound with $\gamma = 4$, i.e.

$$\Pr_{H,\phi}[\rho < 1 - 2^{-t+3}] \le 1/4 \tag{16}$$

Hard to find collisions for H relative to $\mathcal{B}$. We will now show that one cannot find collisions in H even relative to the powerful oracle $\mathcal{B}$.

Lemma 3. *Let (C, P) be as in the statement of Proposition 3 where*

$$v > \log q_P^{\mathcal{B}} + \log q_C + 2(\log(q_P^H + \alpha q_C q_P^{\mathcal{B}})) + 6 \tag{17}$$

and

$$\Pr_{H,\phi,P's\ coins}[\mathsf{col}^H(P^{H,\mathcal{B}})] \ge .675 \tag{18}$$

then H can be compressed below *$2^w v$ bits.*

Before we prove this lemma, we first how it implies Proposition 3.

Proof (of Proposition 3). let $\mathcal{E}$ denote the event that $\mathcal{B}$ ρ-breaks C^H with $\rho \ge 1 - 2^{-t+3}$, using (16) and the $1 - 2^{-t+3}$ security of (C, P) in the last step

$$\Pr_{H,\phi,P's\ coins}[\mathsf{col}^H(P^{\mathcal{B},H})] \ge \Pr_{H,\phi,P's\ coins}[\mathcal{E}] \cdot \Pr_{H,\phi,P's\ coins}[\mathsf{col}^H(P^{\mathcal{B},H})|\mathcal{E}] \ge \frac{3}{4} \cdot 0.9$$

Assume H is uniformly random, then by Lemma 3 the function table of H can be compressed below $2^w v$ bits, which contradicts Proposition 2, thus (17) must be wrong. □

We split the proof of Lemma 3 into two parts. First, Lemma 4 below states that from an oPPTM which finds collisions with high probability as required by eq.(18), we can construct another oPPTM which finds collisions of a special kind, called "very good collisions".[14] Second, Lemma 5 below states that any oPPTM which finds very good collisions for H, implies that H can be compressed.

Very Good Collisions. We now define the "very good collisions" predicate vgCol just mentioned. This predicate has a quite intuitive meaning: $\mathsf{vgCol}(Q^{H,\mathcal{B}})$ if there's a collision, and the H query leading to the collision is fresh, in the sense that it is not in qry_R for some $\mathcal{B}$ query R. More formally, for an oPPTM Q consider the random experiment $Q^{H,\mathcal{B}}$, where $X_1, X_2, \ldots, X_{c_2}$ denotes the H queries, and $R_1, R_2, \ldots, R_j$ denotes the $\mathcal{B}$ queries made by Q. If $\mathsf{col}(Q^{H,\mathcal{B}})$, let X_{c_1}, X_{c_2} denote the collision found by Q. Let qry_R denote all the H queries one must make in order to evaluate $C^H(R, .)$ on the pseudocollision Z_R, Z'_R as sampled by $\mathcal{B}$, i.e.

$$\mathsf{qry}_R := \mathsf{qry}^H(C^H(R, Z_R)) \cup \mathsf{qry}^H(C^H(R, Z'_R)) \tag{19}$$

Then the very good collisions predicate $\mathsf{vgCol}(Q^{H,\mathcal{B}})$ holds if

$$\mathsf{col}(Q^{H,\mathcal{B}}) \text{ and the collision } X_{c_1}, X_{c_2} \text{ satisfies } \forall R \prec X_{c_2} : X_{c_2} \notin \mathsf{qry}_R \tag{20}$$

[13] Unfortunately the I_R's are not independent, thus Chernoff is not an option here.

[14] We leave the term "good collision" for an intermediate kind of collision which will only come up in the proof.

From Good Collisions to Very Good Collisions.

Lemma 4. *If for a PPTM P*

$$\Pr_{H,\phi,P's\ coins}[\mathsf{col}(P^{H,\mathcal{B}})] \geq .675 \tag{21}$$

then there exists a PPTM Q where

$$\Pr_{H,\phi,Q's\ coins}[\mathsf{vgCol}(Q^{H,\mathcal{B}})] \geq .5 \tag{22}$$

and for a constant α

$$q_Q^{\mathcal{B}} = q_P^{\mathcal{B}} \qquad q_Q^H = q_P^H + \alpha q_C q_P^{\mathcal{B}} \tag{23}$$

We omit the proof of this lemma for space reasons. The basic idea of the proof is to let $Q^{H,\mathcal{B}}$ simply simulate $P^{H,\mathcal{B}}$, but whenever P is about to make a $\mathcal{B}$ query R, Q will additionally sample some random $V_1, \ldots, V_\alpha$ and make all the H queries needed to compute $C^H(R, V_i)$. One can show that if the output P gets on his $\mathcal{B}$ query R is likely to contain a collision (which will not be a very good collision), then the H queries Q makes, will also be likely to contain a very good collision. It is the proof of this lemma where we need the fact that $\mathcal{B}(R)$ will output the collision Z_R, Z'_R only if this is a safe collision.

Very Good Collisions Imply Compressibility.

Lemma 5. *Let $\mathcal{B}$ be an oracle (sampled as described earlier in this section) and let H be a random variable taking as values functions $\{0,1\}^w \to \{0,1\}^v$. If a PPTM Q satisfies*

$$\Pr_{H,\phi,Q's\ coins}[\mathsf{vgCol}^H(Q^{\mathcal{B},H})] \geq .5 \tag{24}$$

then for any $0 \leq \gamma \leq 1$, H can be compressed to

$$1 + 2^w v - (1-p)(v - \gamma - 2\log q_Q^H) \tag{25}$$

bits, where $p := 0.5 + q_Q^{\mathcal{B}} \cdot q_C \cdot 2^{-\gamma v}$.

Before we prove this Lemma, let us show how Lemma 4 and 5 imply Lemma 3.

Proof (of Lemma 3). A P as in (18) implies by Lemma 4 a Q as in (22), which by Lemma 5 implies that H can be compressed to (25) bits. This expression is less than $2^w v$ (as required by the lemma) if

$$(0.5 - q_Q^{\mathcal{B}} \cdot q_C \cdot 2^{-\gamma v})(v - \gamma v - 2\log q_Q^H) > 1 \tag{26}$$

By setting $\gamma v := \log q_Q^{\mathcal{B}} + \log q_C + 2$ the first bracket on the left side of (26) becomes $1/4$, if now $v > \log q_Q^{\mathcal{B}} + \log q_C + 2\log q_Q^H + 6 + \log(\ell)$, which by (23) is exactly the requirement (17) from the lemma, then the second bracket in (26) is > 4, thus as $1/4 \cdot 4 = 1$ (26) holds. $\square$

Proof (of Lemma 5). The proof is similar to the proof of Lemma 1, except that now we must additionally handle the breaking oracle $\mathcal{B}$. For this we will additionally need some shared randomness for com and dec, namely a pairwise independent function $\tau : \{0,1\}^w \to \{0,1\}^{\gamma v}$.

If we don't have a very good collision, the compression com simply outputs the whole function table

$$\mathsf{com}(\widetilde{H}) = 0\|\widetilde{H} \quad \texttt{if} \quad \neg\mathsf{vgCol}^H(Q^{\mathcal{B},H})$$

Otherwise let $X_1, X_2, \ldots, X_{c_1}, \ldots, X_{c_2}$ denote the H queries, and $R_1, R_2, \ldots, R_j$ denote the $\mathcal{B}$ queries made by $Q^{\mathcal{B},H}$, where X_{c_1}, X_{c_2} denotes the collision found by Q. With qry_R as defined in (19), let

$$\mathcal{X} := \mathsf{qry}_{R_1} \cup \ldots \cup \mathsf{qry}_{R_j}$$

We define the predicate miss as $\mathsf{miss} \iff \exists X \in \mathcal{X} : \tau(X) = \tau(X_{c_2})$.

The size of $\mathcal{X}$ is upper bounded by $2 \cdot j \cdot q_C \leq 2 \cdot q_Q^{\mathcal{B}} \cdot q_C$. As we now consider the case where $\mathsf{vgCol}^H(Q^{\mathcal{B},H})$, we have $X_{c_2} \notin \mathcal{X}$ (cf. (20)). Further, because τ is pairwise independent, for any $X \neq X_{c_2}$ we have $\mathsf{Pr}[\tau(X) = \tau(X_{c_2})] < 2^{-\gamma v}$. Taking the union bound over all $X \in \mathcal{X}$

$$\begin{aligned}\mathsf{Pr}[\mathsf{miss}] &\leq \mathsf{Pr}[\exists X \in \mathcal{X} : \tau(X_{c_2}) = \tau(X)] \\ &\leq \sum_{X \in \mathcal{X}} \mathsf{Pr}[\tau(X_{c_2)} = \tau(X)] \leq |\mathcal{X}| \cdot 2^{-\gamma v} \leq 2 \cdot q_Q^{\mathcal{B}} \cdot q_C \cdot 2^{-\gamma v} \qquad (27)\end{aligned}$$

In the case where we have miss, com again simply outputs the whole table.

$$\mathsf{com}(\widetilde{H}) = 0\|\widetilde{H} \quad \texttt{if} \quad \mathsf{vgCol}^H(Q^{\mathcal{B},H}) \wedge \mathsf{miss}$$

We will now define an oracle $\mathcal{B}_\tau$, which almost behaves as $\mathcal{B}$, and in particular, whenever we have $\neg\mathsf{miss}$ in $Q^{\mathcal{B},H}$, then the oracle answers of $\mathcal{B}$ in $Q^{\mathcal{B}}$ are identical to the answers of $\mathcal{B}_\tau$ in $Q^{\mathcal{B}_\tau,H}$. Recall that $\mathcal{B}$ on input R samples a pseudocollision Z_R, Z'_R by setting $Z_R := \phi(R)$ and then computes, for $i = 1, 2, \ldots$, the value $Z_i = C^H(R\|\langle i\rangle)$ until $C^H(R, Z_R) = C^H(R, Z_i)$, it then assigns $Z'_R := Z_i$. The oracle $\mathcal{B}_\tau$ does exactly the same, but if the evaluation of $C^H(R, Z_R)$ requires to make an H query X here $\tau(X) = \tau(X_{c_2})$, then $\mathcal{B}_\tau$ does not make this query, but stops and outputs $\perp$. Also, whenever the evaluation of $C^H(R, Z_i)$ requires to make an H query X here $\tau(X) = \tau(X_{c_2})$, then $\mathcal{B}_\tau$ does not make this query, but proceeds with Z_{i+1}. Note that $\mathcal{B}(R)$ and $\mathcal{B}_\tau(R)$ will find the same pseudocollision, iff $\tau(X_{c_2}) \notin \mathsf{qry}_R$.

Recall that we now only consider the case where $\mathsf{vgCol}^H(Q^{\mathcal{B},H})$ and $\neg\mathsf{miss}$. Consider the random experiment $Q^{\mathcal{B}_\tau,H}$, and let $A_1, A_2, \ldots, A_\sigma$, where $A_\sigma = X_{c_2}$ denote all the H queries done by Q plus the H queries made by $\mathcal{B}_\tau$ (in the order as they are made in the random experiment $Q^{\mathcal{B}_\tau,H}$ up to the "collision finding" H query X_{c_2}, but without repetitions). So each H query by Q increases the sequence $A_1, A_2, \ldots$ at most by one, whereas a $\mathcal{B}_\tau$ query by Q can increase it by arbitrary many values. Let $\widetilde{H}^-$ denote the function table of H, but with the rows $A_1, A_2, \ldots, A_\sigma$ deleted. The compression algorithm com for the remaining cases is now defined as

$$\mathsf{com}(\widetilde{H}) = 1\|\tau(X_{c_2})\|\langle c_1\rangle_{\log_{q_Q}}\|\langle c_2\rangle_{\log_{q_Q}}\|H(A_1)\|\ldots\|H(A_{\sigma-1})\|\widetilde{H}^- \tag{28}$$

$$\text{if} \quad \mathsf{vgCol}^H(Q^{\mathcal{B},H}) \wedge \neg\mathsf{miss} \tag{29}$$

Let us check that this compression really compresses as claimed by eq. (25). If $\neg\mathsf{vgCol}^H(Q^{\mathcal{B},H})$, or $\mathsf{vgCol}^H(Q^{\mathcal{B},H}) \wedge \mathsf{miss}$ then then $|\mathsf{com}(\widetilde{H})| = 2^w v + 1$. By (24),(27) this happens with probability at most $p := 0.5 + 0.5(2 \cdot q_Q^{\mathcal{B}} \cdot q_C \cdot 2^{-\gamma v})$. Otherwise by (29) $|\mathsf{com}(\widetilde{H})|$ has length only $1 + (2^w - 1 + \gamma)v + 2\log q_Q^H$. Thus

$$\mathsf{E}[|\mathsf{com}(\widetilde{H})|] \leq 1 + 2^w v - (1-p)(v - \gamma - 2\log q_Q^H)$$

as required by (25). Decompression is straight forward and omitted for space reasons. $\square$

Lower bound for (k,ℓ)-combiners. In the full version of this paper [15] we prove the following Proposition which implies Theorem 1 for general (k,ℓ)-combiners.

Proposition 4. *Let $C : \{0,1\}^r \times \{0,1\}^m \to \{0,1\}^n$ be an oracle circuit with input range*[15] *$m := \ell \cdot (v+1)$ bits and q_C oracle gates, where for some $t > 0$*

$$n := (\ell - k + 1) \cdot (v - 2\log(q_C)) - t$$

then, if for some oracle PPTM P which makes $q_P^{\mathcal{B}}$ oracle calls to the breaking oracle and q_P^H oracle calls to its components it is the case that (C,P) is a ρ-robust (k,ℓ)-combiner with $\rho := 1/\binom{\ell}{k} - 2^{-t+\ell+2}$, then

$$v \leq \log q_P^{\mathcal{B}} + \log q_C + 2(\log(q_P^H + \alpha q_C q_P^{\mathcal{B}})) + 6 + \log(\ell) \tag{30}$$

in particular, (C,P) is not efficient.

Acknowledgements

I'd like to thank the anonymous reviewers from Crypto'08 for their many helpful comments and suggestions.

References

1. Boaz Barak. How to go beyond the black-box simulation barrier. In *42nd FOCS*, pages 106–115. IEEE Computer Society Press, October 2001.
2. Dan Boneh and Xavier Boyen. On the impossibility of efficiently combining collision resistant hash functions. In *CRYPTO 2006*, volume 4117 of *LNCS*, pages 570–583. August 2006.

[15] Remark 4 applies here too.

3. Ran Canetti, Ronald L. Rivest, Madhu Sudan, Luca Trevisan, Salil P. Vadhan, and Hoeteck Wee. Amplifying collision resistance: A complexity-theoretic treatment. In *CRYPTO 2007*, volume 4622 of *LNCS*, pages 264–283. August 2007.
4. Marc Fischlin and Anja Lehmann. Security-amplifying combiners for collision-resistant hash functions. In *CRYPTO 2007*, volume 4622 of *LNCS*, pages 224–243. August 2007.
5. Rosario Gennaro and Luca Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *41st FOCS*, pages 305–313. IEEE Computer Society Press, November 2000.
6. Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the ACM*, 38(3):691–729, 1991.
7. Iftach Haitner, Jonathan J. Hoch, Omer Reingold, and Gil Segev. Finding collisions in interactive protocols - a tight lower bound on the round complexity of statistically-hiding commitments. In *48th FOCS*, pages 669–679. IEEE Computer Society Press, October 2007.
8. Danny Harnik, Joe Kilian, Moni Naor, Omer Reingold, and Alon Rosen. On robust combiners for oblivious transfer and other primitives. In *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 96–113. May 2005.
9. Amir Herzberg. On tolerant cryptographic constructions. In *CT-RSA 2005*, volume 3376 of *LNCS*, pages 172–190. February 2005.
10. Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *21st ACM STOC*, pages 44–61. ACM Press, May 1989.
11. Jeong Han Kim, Daniel R. Simon, and Prasad Tetali. Limits on the efficiency of one-way permutation-based hash functions. In *40th FOCS*, pages 535–542. IEEE Computer Society Press, October 1999.
12. Anja Lehmann Marc Fischlin and Krzysztof Pietrzak. Robust multi-property combiners for hash functions revisited. In *ICALP 2008*, LNCS. 2008.
13. Remo Meier and Bartosz Przydatek. On robust combiners for private information retrieval and other primitives. In *CRYPTO 2006*, volume 4117 of *LNCS*, pages 555–569. August 2006.
14. Remo Meier, Bartosz Przydatek, and Jürg Wullschleger. Robuster combiners for oblivious transfer. In *TCC 2007*, volume 4392 of *LNCS*, pages 404–418. February 2007.
15. Krzysztof Pietrzak. Full version at www.cwi.nl/∼pietrzak/publications.html.
16. Krzysztof Pietrzak. Non-trivial black-box combiners for collision-resistant hash-functions don't exist. In *EUROCRYPT 2007*, volume 4515 of *LNCS*, pages 23–33. May 2007.
17. Omer Reingold, Luca Trevisan, and Salil P. Vadhan. Notions of reducibility between cryptographic primitives. In *TCC 2004*, volume 2951 of *LNCS*, pages 1–20. February 2004.
18. Phillip Rogaway. Formalizing human ignorance. In *Progress in Cryptology - VIETCRYPT 06*, volume 4341 of *LNCS*, pages 211–228. September 2006.
19. Daniel R. Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In *EUROCRYPT'98*, volume 1403 of *LNCS*, pages 334–345. May / June 1998.
20. Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu. Finding collisions in the full SHA-1. In *CRYPTO 2005*, volume 3621 of *LNCS*, pages 17–36. August 2005.
21. Xiaoyun Wang and Hongbo Yu. How to break MD5 and other hash functions. In *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 19–35. May 2005.