Multi-Collision Resistant Hash Functions and their Applications*

Itay Berman[1], Akshay Degwekar[1], Ron D. Rothblum[12], and Prashant Nalini Vasudevan[1]

[1] MIT
{itayberm,akshayd,prashvas,ronr}@mit.edu
[2] Northeastern University

Abstract. Collision resistant hash functions are functions that shrink their input, but for which it is computationally infeasible to find a collision, namely two strings that hash to the same value (although collisions are abundant).
In this work we study *multi-collision resistant hash functions* (MCRH) a natural relaxation of collision resistant hash functions in which it is difficult to find a t-way collision (i.e., t strings that hash to the same value) although finding $(t-1)$-way collisions could be easy. We show the following:

- The existence of MCRH follows from the average case hardness of a variant of the *Entropy Approximation* problem. The goal in this problem (Goldreich, Sahai and Vadhan, CRYPTO '99) is to distinguish circuits whose output distribution has high entropy from those having low entropy.
- MCRH imply the existence of *constant-round* statistically hiding (and computationally binding) commitment schemes. As a corollary, using a result of Haitner et al. (SICOMP, 2015), we obtain a blackbox separation of MCRH from any one-way permutation.

1 Introduction

Hash functions are efficiently computable functions that shrink their input and mimic 'random functions' in various aspects. They are prevalent in cryptography, both in theory and in practice. A central goal in the study of the foundations of cryptography has been to distill the precise, and minimal, security requirements necessary from hash functions for different applications.

One widely studied notion of hashing is that of *collision resistant hash functions* (CRH). Namely, hash functions for which it is computationally infeasible to find two strings that hash to the same value, even when such collisions are abundant. CRH have been extremely fruitful and have notable applications in cryptography such as digital signatures[3] [GMR88], efficient argument systems for

[*] The full version [BDRV17] is available at https://eprint.iacr.org/2017/489.

[3] We remark that the weaker notion of universal one-way hash functions (UOWHF) (which is known to be implied by standard one-way functions) suffices for this application [NY89, Rom90].

NP [Kil92, Mic00] and (constant-round) statistically hiding commitment schemes [NY89, DPP93, HM96].

In this work we study a natural relaxation of collision resistance. Specifically, we consider hash functions for which it is infeasible to find a t-way collision: i.e., t strings that all have the same hash value. Here t is a parameter, where the standard notion of collision resistance corresponds to the special case of $t = 2$. We refer to such functions as *multi-collision resistant hash functions (*MCRH*)* and emphasize that, for $t > 2$, it is a *weaker* requirement than that of standard collision resistance.

The property of multi-collision resistance was considered first by Merkle [Mer89] in analyzing a hash function construction based on DES. The notion has also been considered in the context of identification schemes [GS94], micropayments [RS96], and signature schemes [BPVY00]. Joux [Jou04] showed that for *iterated* hash functions, finding a large number of collisions is no harder than finding pairs of highly structured colliding inputs (namely, collisions that share the same prefix). We emphasize that Joux's multi-collision finding attack only applies to certain types of hash functions (e.g., iterated hash functions, or tree hashing) and requires a strong break of collision resistance. In general, it seems that MCRH is a weaker property than CRH.

As in the case of CRH, to obtain a meaningful definition, we must consider keyed functions (since for non keyed functions there are trivial non-uniform attacks). Thus, we define MCRH as follows (here and throughout this work, we use n to denote the security parameter.)

Definition 1.1 ((s,t)-MCRH). *Let $s = s(n) \in \mathbb{N}$ and $t = t(n) \in \mathbb{N}$ be functions computable in time* $\mathsf{poly}(n)$*. An (s,t)-*Multi-Collision Resistant Hash Function Family ((s,t)-MCRH) *consists of a probabilistic polynomial-time algorithm* Gen *that on input 1^n outputs a circuit h such that:*

- **s-Shrinkage:** *The circuit $h : \{0,1\}^n \to \{0,1\}^{n-s}$ maps inputs of length n to outputs of length $n - s$.*
- **t-Collision Resistance:** *For every polynomial size family of circuits $\mathsf{A} = (\mathsf{A}_n)_{n\in\mathbb{N}}$,*

$$\Pr_{\substack{h\leftarrow\mathsf{Gen}(1^n),\\ (x_1,x_2,\dots,x_t)\leftarrow\mathsf{A}_n(h)}}\left[\begin{matrix}\text{For all } i \neq j,\\ h(x_i) = h(x_j) \text{ and } x_i \neq x_j\end{matrix}\right] < \mathsf{negl}(n).$$

Note that the standard notion of CRH simply corresponds to $(1,2)$-MCRH (which is easily shown to be equivalent to $(s,2)$-CRH for any $s = n - \omega(\log n)$). We also remark that Definition 1.1 gives a *non-uniform* security guarantee, which is natural, especially in the context of collision resistance. Note though that all of our results are obtained by *uniform* reductions.

Remark 1.2 (Shrinkage vs. Collision Resistance). Observe that (s,t)-MCRH are meaningful only when $s \geq \log t$, as otherwise t-way collisions might not even exist (e.g., consider a function mapping inputs of length n to outputs of length $n - \log(t-1)$ in which each range element has exactly $t - 1$ preimages).

Moreover, we note that in contrast to standard CRH, it is unclear whether the shrinkage factor s can be trivially improved (e.g., by composition) while preserving the value of t. Specifically, constructions such as Tree Hashing (aka Merkle Tree) inherently rely on the fact that it is computationally infeasible to find *any* collision. It is possible to get some trade-offs between the number of collisions and shrinkage. For example, given an $(s = 2, t = 4)$-MCRH, we can compose it with itself to get an $(s = 4, t = 10)$-MCRH. But it is not a priori clear whether there exist transformations that increase the shrinkage s while not increasing t. We remark that a partial affirmative answer to this question was recently given in an independent and concurrent work by Bitansky et al. [BPK17], as long as the hash function is substantially shrinking (see additional details in Section 1.2).

Thus, we include both the parameters s and t in the definition of MCRH, whereas in standard CRH the parameter t is fixed to 2, and the parameter s can be given implicitly (since the shrinkage can be trivially improved by composition).

Remark 1.3 (Scaling of Shrinkage vs. Collisions). The shrinkage s is measured in bits, whereas the number of collisions t is just a number. A different definitional choice could have been to put s and t on the same "scale" (e.g., measure the logarithm of the number of collisions) so to make them more easily comparable. However, we refrain from doing so since we find the current (different) scaling of s and t to be more natural.

Remark 1.4 (Public-coin MCRH*).* One can also consider the stronger *public-coin* variant of MCRH, in which it should be hard to find collisions given not only the description of the hash function, but also the coins that generated the description.

Hsiao and Reyzin [HR04] observed that for some applications of standard collision resistance, it is vital to use the public-key variant (i.e., security can be broken in case the hash function is not public-coin). The distinction is similarly important for MCRH and one should take care of which notion is used depending on the application. Below, when we say MCRH, we refer to the *private-coin* variant (as per Definition 1.1).

1.1 Our Results

The focus of this work is providing a systematic study of MCRH. We consider both the question of constructing MCRH and what applications can we derive from them.

1.1.1 Constructions of MCRH

Since any CRH is in particular also an MCRH, candidate constructions are abundant (based on a variety of concrete computational assumptions). The actual question that we ask, which has a more foundational flavor, is whether we can construct MCRH from assumptions that are not known to imply CRH.

Our first main result is that the existence of MCRH follows from the average-case hardness of a variant of the *Entropy Approximation* problem studied by Goldreich, Sahai and Vadhan [GSV99]. Entropy Approximation, denoted EA, is a promise problem, where YES inputs are circuits whose output distribution (i.e., the distribution obtained by feeding random inputs to the circuit) has entropy at least k, whereas NO inputs are circuits whose output distribution has entropy at most $k-1$ (where k is a parameter that is unimportant for the current discussion). Here by entropy we specifically refer to *Shannon* entropy.[4] Goldreich et al. showed that EA is complete for the class of (promise) problems that have non-interactive statistical zero-knowledge proofs (NISZK).

In this work we consider a variant of EA, first studied by Dvir et al. [DGRV11], that uses different notions of entropy. Specifically, consider the promise problem $\mathsf{EA}_{\mathrm{min,max}}$, where the goal now is to distinguish between circuits whose output distribution has *min*-entropy[5] at least k from those with *max*-entropy at most $k-1$. It is easy to verify that $\mathsf{EA}_{\mathrm{min,max}}$ is an easier problem than EA.

Theorem 1.1 (Informal, see Theorem 3.6). *If $\mathsf{EA}_{\mathrm{min,max}}$ is average-case hard, then there exist (s,t)-MCRH, where $s=\sqrt{n}$ and $t=6n^2$.*

(Note that in the MCRH that we construct there exist $2^{\sqrt{n}}$-way collisions, but it is computationally hard to find even a $6n^2$-way collision.)

In contrast to the original entropy approximation problem, we do not know whether $\mathsf{EA}_{\mathrm{min,max}}$ is complete for NISZK. Thus, establishing the existence of MCRH based solely on the average-case hardness of NISZK (or SZK) remains open. Indeed such a result could potentially be an interesting extension of Ostrovsky's [Ost91] proof that average-case hardness of SZK implies the existence of one-way functions.

Instantiations. Dvir et al. [DGRV11], showed that the average-case hardness of $\mathsf{EA}_{\mathrm{min,max}}$ is implied by either the quadratic residuocity (QR) or decisional Diffie Hellman (DDH) assumptions.[6] It is not too hard to see that above extends to any encryption scheme (or even commitment scheme) in which ciphertexts can be perfectly re-randomized.[7]

[4] Recall that the Shannon Entropy of a random variable X is defined as $\mathrm{H}_{\mathrm{Shannon}}(X)=\mathrm{E}_{x\leftarrow X}\left[\log\left(\frac{1}{\Pr[X=x]}\right)\right]$.

[5] For a random variable X, the min-entropy is defined as $\mathrm{H}_{\mathrm{min}}(X) = \min_{x\in\mathrm{Supp}(X)}\log\left(\frac{1}{\Pr[X=x]}\right)$ whereas the max-entropy is $\mathrm{H}_{\mathrm{max}}(X)=\log\left(|\mathrm{Supp}(X)|\right)$.

[6] In fact, [DGRV11] show that the same conclusion holds even if we restrict the problem to constant-depth (i.e., NC_0) circuits.

[7] Given such a scheme consider a circuit that has, hard-coded inside, a pair of ciphertexts (c_0,c_1) which are either encryptions of the same bit or of different bits. The circuit gets as input a bit b and random string r and outputs a re-randomization of c_b (using randomness r). If the scheme is perfectly re-randomizing (and perfectly correct) then the min-entropy of the output distribution in case the plaintexts disagree is larger than the max-entropy in case the plaintexts agree.

The hardness of $\mathsf{EA}_{\min,\max}$ can also be shown to follow from the *average-case* hardness of the Shortest Vector Problem or the Closest Vector Problem with approximation factor roughly $\sqrt{n}$.[8] To the best of our knowledge the existence of CRH is not known based on such small approximation factors (even assuming average-case hardness).

We remark that a similar argument establishes the hardness of $\mathsf{EA}_{\min,\max}$ based on the plausible assumption that graph isomorphism is average-case hard.[9]

1.1.2 Applications of MCRH

The main application that we derive from MCRH is a *constant-round* statistically hiding commitment scheme.

Theorem 1.2 (Informally stated, see Theorem 4.4). *Assume that there exists a* $(\log(t), t)$-MCRH*. Then, there exists a* 3*-round statistically-hiding and computationally-binding commitment scheme.*

We note that Theorem 1.2 is optimal in the sense of holding for MCRH that are minimally shrinking. Indeed, as noted in Remark 1.2, (s, t)-MCRH with $s \leq \log(t-1)$ exist trivially and unconditionally.

It is also worthwhile to point out that by a result of Haitner et al. [HNO$^+$09], statistically-hiding commitment schemes can be based on the existence of any one-way function. However, the commitment scheme of [HNO$^+$09] uses a polynomial number of rounds of interaction and the main point in Theorem 1.2 is that we obtain such a commitment scheme with only a *constant* number of rounds.

Moreover, by a result of [HHRS15], any *fully black-box* construction of a statistically hiding commitment scheme from one-way functions (or even one-way permutations) must use a polynomial number of rounds. Loosely speaking, a construction is "fully black-box" [RTV04] if (1) the construction only requires an input-output access to the underlying primitive and (2) the security proof also

[8] The hard distribution for $\mathsf{SVP}_{\sqrt{n}}$ and $\mathsf{CVP}_{\sqrt{n}}$ is the first message from the 2-message honest-verifier SZK proof system of Goldreich and Goldwasser [GG98]. In the case of $\mathsf{CVP}_{\sqrt{n}}$, the input is (B, t, d) where B is the basis of the lattice, t is a target vector and d specifies the bound on the distance of t from the lattice. The distribution is obtained by sampling a random error vector η from the ball of radius $d\sqrt{n}/2$ centered at the origin and outputting $b \cdot t + \eta \mod \mathcal{P}(B)$, where $b \leftarrow \{0,1\}$ and $\mathcal{P}(B)$ is the fundamental parallelopiped of B. When t is far from the lattice, this distribution is injective and hence has high min-entropy while when t is close to the lattice, the distribution is not injective and hence has lower max-entropy. Similarly for $\mathsf{SVP}_{\sqrt{n}}$, on input (B, d), the output is $\eta \mod \mathcal{P}(B)$ where η is again sampled from a ball of radius $d\sqrt{n}/2$.

[9] Note that the graph isomorphism is known to be solvable in polynomial-time for many natural distributions, and the recent breakthrough result of Babai [Bab16] gives a quasi-polynomial worst-case algorithm. Nevertheless, it is still plausible that Graph Isomorphism is average-case quasi-polynomially hard (for some efficiently samplable distribution).

relies on the adversary in a black-box way. Most constructions in cryptography are fully black-box. Since our proof of Theorem 1.2 is via a fully black-box construction, we obtain the following immediate corollary:

Corollary 1.3 (Informally stated) *There does not exist a fully blackbox construction of* MCRH *from one-way permutations.*

Corollary 1.3 can be viewed as an extension of Simon's [Sim98] blackbox separation of CRH from one-way permutations. Due to space limitations, the formal statement and proof of Corollary 1.3 is deferred to the full version of this paper [BDRV17].

1.2 Related Works

Generic Constructions of CRH. Peikert and Waters [PW11] construct CRH from lossy trapdoor functions. Their construction can be viewed as a construction of CRH from $\mathsf{EA}_{\mathrm{min,max}}$ with a huge gap. (Specifically, the lossy trapdoor function h is either injective (i.e., $\mathrm{H}_{\mathrm{min}}(h) \geq n$) or very shrinking (i.e., $\mathrm{H}_{\mathrm{max}}(h) < 0.5n$).[10] One possible approach to constructing CRH from lossy functions with small 'lossiness' ($\mathrm{H}_{\mathrm{max}}(h)/\,\mathrm{H}_{\mathrm{min}}(h)$) is to first amplify the lossiness and then apply the [PW11] construction. Pietrzak et al. [PRS12] rule out this approach by showing that it is impossible to improve the 'lossiness' in a black-box way.[11] We show that even with distributions where the gap is tiny, we can achieve weaker yet very meaningful notions of collision-resistance.

Applebaum and Raykov [AR16] construct CRH from any average-case hard language with a *Perfect Randomized Encoding* in which the encoding algorithm is one-to-one as a function of the randomness. Perfect Randomized Encodings are a way to encode the computation of a function f on input x such that information-theoretically, the *only* information revealed about x is the value $f(x)$. The class of languages with such randomized encodings PRE is contained in PZK. Their assumption of an average-case hard language with a perfect randomized encoding implies $\mathsf{EA}_{\mathrm{min,max}}$ as well.

Constant-Round Statistically Hiding Commitments from SZK Hardness. The work of Ong and Vadhan [OV08] yields constant-round statistically-hiding commitment schemes from average-case hardness of SZK.[12] Our construction of statistically-hiding commitments via MCRH is arguably simpler, although it relies on a stronger assumption ($\mathsf{EA}_{\mathrm{min,max}}$) instead of average-case hardness of SZK.

[10] The trapdoor to the lossy function is not used in the construction of CRH.

[11] In contrast, it is easy to see that repetition amplifies the *additive* gap between the min-entropy and the max-entropy. In fact, we use this in our construction.

[12] Actually, Ong and Vadhan [OV08] only construct instance-dependent commitments. Dvir et al. [DGRV11] attribute the construction of constant-round statistically hiding commitments from average-case hardness of SZK to a combination of [OV08] and an unpublished manuscript of Guy Rothblum and Vadhan [RV09].

Distributional CRH. A different weakening of collision resistance was considered by Dubrov and Ishai [DI06]. Their notion, called "distributional collision-resistant" in which it may be feasible to find some specific collision, but it is hard to sample a *random* collision pair. That is, given the hash function h, no efficient algorithm can sample a pair (z_1, z_2) such that z_1 is uniform and z_2 is uniform in the set $\{z : h(z) = h(z_1)\}$. The notions of MCRH and distributional CRH are incomparable and whether one can be constructed from the other is open.

Min-Max Entropy Approximation. The main result of the work of Dvir et al. [DGRV11] (that was mentioned above) was showing that the problem EA for degree-3 polynomial mappings (i.e., where the entropies are measured by Shannon entropy) is complete for $\mathsf{SZK_L}$, a sub-class of SZK in which the verifier and the simulator run in logarithmic space. They also construct algorithms to approximate different notions of entropy in certain restricted settings (but their algorithms do not violate the assumption that $\mathsf{EA}_{\min,\max}$ is average-case hard).

1.2.1 Independent Works

MCRH have been recently considered in an independent work by Komargodski et al. [KNY17b] (which was posted online roughly four months prior to the first public posting of our work). Komargodski et al. study the problem, arising from Ramsey theory, of finding either a clique or an independent set (of roughly logarithmic size) in a graph, when such objects are guaranteed to exist. As one of their results, [KNY17b] relate a variant of the foregoing Ramsey problem (for bipartite graphs) to the existence of MCRH. We emphasize that the focus of [KNY17b] is in studying computational problems arising from Ramsey theory, rather than MCRH directly.

Beyond the work of [KNY17b], there are two other concurrent works that specifically study MCRH [BPK17, KNY17a] (and were posted online simultaneously to our work). The main result of [KNY17a] is that the existence of MCRH (with suitable parameters) implies the existence of efficient argument-systems for NP, *á* la Kilian's protocol [Kil92]. Komargodski et al. [KNY17a] also prove that MCRH imply constant-round statistically hiding commitments (similarly to Theorem 1.2), although their result only holds for MCRH who shrink their input by a constant multiplicative factor. Lastly, [KNY17a] also show a blackbox separation between MCRH in which it is hard to find t collisions from those in which it is hard to find $t + 1$ collisions.

Bitansky et al. [BPK17] also study MCRH, with the motivation of constructing efficient argument-systems. They consider both a keyed version of MCRH (as in our work) and an unkeyed version (in which, loosely speaking, the requirement is that adversary cannot produce more collisions than those it can store as non-uniform advice). [BPK17] show a so-called "domain extension" result for MCRH that are sufficiently shrinking. Using this result they construct various succinct and/or zero-knowledge argument-systems, with optimal or close-to-optimal round complexity. In particular, they show the existence of 4 round zero-knowledge arguments for NP based on MCRH, and, assuming unkeyed MCRH, they obtain a similar result but with only 3 rounds of interaction.

1.3 Our Techniques

We provide a detailed overview of our two main results: Constructing MCRH from $\mathsf{EA}_{\min,\max}$ and constructing constant-round statistically-hiding commitment scheme from MCRH.

1.3.1 Constructing MCRH from $\mathsf{EA}_{\mathbf{min,max}}$

Assume that we are given a distribution on circuits $\left\{C\colon \{0,1\}^n \to \{0,1\}^{2n}\right\}$ such that that it is hard to distinguish between the cases $\mathrm{H}_{\min}(C) \geq k$ or $\mathrm{H}_{\max}(C) \leq k-1$, where we overload notation and let C also denote the output distribution of the circuit when given uniformly random inputs. Note that we have set the output length of the circuit C to $2n$ but this is mainly for concreteness (and to emphasize that the circuit need not be shrinking).

Our goal is to construct an MCRH using C. We will present our construction in steps, where in the first case we start off by assuming a very large entropy gap. Specifically, for the first (over-simplified) case, we assume that it is hard to distinguish between min-entropy $\geq n$ vs. max-entropy $\leq n/2$.[13] Note that having min-entropy n means that C is *injective.*

Warmup: The case of $\mathrm{H}_{\min}(C) \geq n$ vs. $\mathrm{H}_{\max}(C) \ll n/2$. In this case, it is already difficult to find even a 2-way collision in C: if $\mathrm{H}_{\min}(C) \geq n$, then C is injective and no collisions exist. Thus, if one can find a collision, it must be the case that $\mathrm{H}_{\max}(C) \leq n/2$ and so any collision finder distinguishes the two cases.

The problem though is that C by itself is not shrinking, and thus is not an MCRH. To resolve this issue, a natural idea that comes to mind is to hash the output of C, using a pairwise independent hash function.[14] Thus, the first idea is to choose $f : \{0,1\}^{2n} \to \{0,1\}^{n-s}$, for some $s \geq 1$, from a family of pairwise independent hash functions and consider the hash function $h(x) = f(C(x))$.

If $\mathrm{H}_{\min}(C) \geq n$ (i.e., C is injective), then every collision in h is a collision on the hash function f. On the other hand, if $\mathrm{H}_{\max}(C) \leq n/2$, then C itself has many collisions. To be able to distinguish between the two cases, we would like that in the latter case there will be no collisions that originate from f. The image size of C, if $\mathrm{H}_{\max}(C) \ll n/2$, is smaller than $2^{n/2}$. If we set s to be sufficiently small (say constant) than the range of f has size roughly 2^n. Thus, we are hashing a set into a range that is more than quadratic in its size. In such case, we are "below the birthday paradox regime" and a *random function* on this

[13] This setting (and construction) is similar to that of Peikert and Waters's construction of CRH from lossy functions [PW11].

[14] Recall that a collection of functions $\mathcal{F}$ is k-wise independent if for every distinct $x_1, \ldots, x_k$, the distribution of $(f(x_1), \ldots, f(x_k))$ (over the choice of $f \leftarrow \mathcal{F}$) is uniform.

set will be injective. A similar statement can be easily shown also for functions that are merely pairwise independent (rather than being entirely random).

Thus, in case C is injective, all the collisions appear in the second part of the hash function (i.e., the application of f). On the other hand, if C has max-entropy smaller than $n/2$, then all the collisions happen in the first part of the hash function (i.e., in C). Thus, any adversary that finds a collision distinguishes between the two cases and we actually obtain a full-fledged CRH (rather than merely an MCRH) at the cost of making a much stronger assumption.

The next case that we consider is still restricted to circuits that are injective (i.e., have min entropy n) in one case but assumes that it is hard to distinguish injective circuits from circuits having max-entropy $n-\sqrt{n}$ (rather than $n/2$ that we already handled).

The case of $\mathrm{H}_{\min}(C) \geq n$ vs. $\mathrm{H}_{\max}(C) \leq n - \sqrt{n}$. The problem that we encounter now is that in the low max entropy case, the output of C has max-entropy $n-\sqrt{n}$. To apply the above birthday paradox argument we would need the range of f to be of size roughly $(2^{n-\sqrt{n}})^2 \gg 2^n$ and so our hash function would not be shrinking. Note that if the range of f were smaller, than even if f were chosen entirely at random (let alone from a pairwise independent family) we would see collisions in this case (again, by the birthday paradox).

The key observation that we make at this point is that although we will see collisions, there will not be too many of them. Specifically, suppose we set $s \approx \sqrt{n}$. Then, we are now hashing a set of size $2^{n-\sqrt{n}}$ into a range of size $2^{n-\sqrt{n}}$. If we were to choose f entirely at random, this process would correspond to throwing $N = 2^{n-\sqrt{n}}$ balls (i.e., the elements in the range of C) into N bins (i.e., elements in the range of f). It is well-known that in such case, with high probability, the maximal load for any bin will be at most $\frac{\log(N)}{\log\log(N)} < n$. Thus, we are guaranteed that there will be at most n collisions.

Unfortunately, the work of Alon et al. [ADM+99] shows that the same argument does not apply to functions that are merely pairwise independent (rather than entirely random). Thankfully though, suitable derandomizations are known. Specifically, it is not too difficult to show that if we take f from a family of *n-wise independent hash functions*, then the maximal load will also be at most n (see Section 2.2 for details).[15]

Similarly to before, in case C is injective, there are no collisions in the first part. On the other hand, in case C has max-entropy at most $n-\sqrt{n}$, we have just argued that there will be less than n collisions in the second part. Thus, an adversary that finds an n-way collision distinguishes between the two cases and we have obtained an (s,t)-MCRH, with $s = \sqrt{n}$ and $t = n$ (i.e., collisions of size $2^{\sqrt{n}}$ exist but finding a collision of size even n is computationally infeasible).

The case of $\mathrm{H}_{\min}(C) \geq k$ vs. $\mathrm{H}_{\max}(C) \leq k - \sqrt{n}$. We want to remove the assumption that when the min-entropy of C is high, then it is in fact injective.

[15] We remark that more efficient constructions are known, see Remark 2.4.

Specifically, we consider the case that either C's min-entropy is at least k (for some parameter $k \leq n$) or its max entropy is at most $k - \sqrt{n}$. Note that in the high min-entropy case, C — although not injective — maps at most 2^{n-k} inputs to every output (this is essentially the definition of min-entropy). Our approach is to apply hashing a second time (in a different way), to effectively make C injective, and then apply the construction from the previous case.

Consider the mapping $h'(x) = (C(x), f(x))$, where f will be defined ahead. For h' to be injective, f must be injective over all sets of size 2^{n-k}. Taking f to be pairwise-independent will force to set its output length to be too large, in a way that will ruin the entropy gap between the cases.

As in the previous case, we resolve this difficulty by using many-wise independent hashing. Let $f\colon \{0,1\}^n \to \{0,1\}^{n-k}$ be a $3n$-wise independent hash function. If $\mathrm{H}_{\min}(C) \geq k$, then the same load-balancing property of f that we used in the previous case, along with a union bound, implies that with high probability (over the choice of f) there will be no $3n$-way collisions in h'. Our final construction applies the previous construction on h'. Namely,

$$h_{C,f,g}(x) = g(C(x), f(x)),$$

for $f\colon \{0,1\}^n \to \{0,1\}^{n-k}$ and $g\colon \{0,1\}^{3n-k} \to \{0,1\}^{n-\sqrt{n}}$ being $3n$-wise and $2n$-wise independent hash functions, respectively. We can now show that

- If $\mathrm{H}_{\min}(C) \geq k$, then there do not exist $3n$ distinct inputs $x_1, \ldots, x_{3n}$ such that they all have the same value of $(C(x_i), f(x_i))$; and
- If $\mathrm{H}_{\max}(C) \leq k - \sqrt{n}$, then there do not exist $2n$ distinct inputs $x_1, \ldots, x_{2n}$ such that they all have distinct values of $(C(x_i), f(x_i))$, but all have the same value $g(C(x_i), f(x_i))$.

We claim that $h_{C,f,g}$ is (s,t)-MCRH for $s = \sqrt{n}$ and $t = 6n^2$: First, note that in any set of $6n^2$ collisions for $h_{C,f,g}$, there has to be either a set of $3n$ collisions for (C, f) or a set of $2n$ collisions for g, and so at least one of the conditions in the above two statements is violated. Now, assume that an adversary A finds a $6n^2$-way collision in $h_{C,f,g}$ with high probability. Then, an algorithm D that distinguishes between $\mathrm{H}_{\min}(C) \geq k$ to $\mathrm{H}_{\max}(C) \leq k - \sqrt{n}$ chooses f and g uniformly at random and runs A on the input $h = h_{C,f,g}$ to get $x_1, \ldots, x_{6n^2}$ with $h(x_1) = \cdots = h(x_{6n^2})$. The distinguisher D now checks which of the two conditions above is violated, and thus can distinguish if it was given C with $\mathrm{H}_{\min}(C) \geq k$ or $\mathrm{H}_{\max}(C) \leq k - \sqrt{n}$.

We proceed to the case that the entropy gap is 1 (rather than $\sqrt{n}$). This case is rather simple to handle (via a reduction to the previous case).

The case of $\mathrm{H}_{\min}(C) \geq k$ vs. $\mathrm{H}_{\max}(C) \leq k - 1$. This case is handled by reduction to the previous case. The main observation is that if C has min-entropy at least k, and we take ℓ copies of C, then we get a new circuit with min-entropy at least $\ell \cdot k$. In contrast, if C had max-entropy at most $k-1$, then

C' has max-entropy at most $\ell \cdot k - \ell$. Setting $\ell = k$, we obtain that in the second case the max-entropy is $n' - \sqrt{n'}$, where $n' = \ell \cdot k$ is the new input length. Thus, we have obtained a reduction to the $\sqrt{n'}$ gap case that we already handled.

1.3.2 Statistically-Hiding Commitments from MCRH

The fact that MCRH imply constant-round statistically-hiding commitments can be shown in two ways. The first, more direct way, uses only elementary notions such as k-wise independent hashing and is similar to the interactive hashing protocol of Ding et al. [DHRS07]. An alternative method, is to first show that MCRH imply the existence of an ($O(1)$-block) *inaccessible entropy generator* [HRVW09, HV17]. The latter was shown by [HRVW09, HV17] to imply the existence of constant-round statistically-hiding commitments. We discuss these two methods next and remark that in our actual proof we follow the direct route.

1.3.2.1 Direct Analysis

In a nutshell our approach is to follow the construction of Damgård et al. [DPP93] of statistically-hiding commitments from CRH, while replacing the use of pairwise independent hashing, with the interactive hashing of Ding et al. [DHRS07]. We proceed to the technical overview, which does not assume familiarity with any of these results.

Warmup: Commitment from (Standard) CRH. Given a family of collision resistant hash functions $\mathcal{H} = \left\{h \colon \{0,1\}^n \to \{0,1\}^{n-1}\right\}$, a natural first attempt is to have the receiver sample the hash function $h \leftarrow \mathcal{H}$ and send it to the sender. The sender, trying to commit to a bit b, chooses $x \leftarrow \{0,1\}^n$ and $r \leftarrow \{0,1\}^n$, and sends $(y = h(x), r, \sigma = \langle r, x\rangle \oplus b)$ to the receiver. The commitment is defined as $c = (h, y, r, \sigma)$. To reveal, the sender sends (x, b) to the receiver, which verifies that $h(x) = y$ and $\sigma = \langle r, x\rangle \oplus b$. Pictorially, the commit stage is as follows:

$$\begin{tikzcd}
\mathsf{S}(b) & & & \mathsf{R} \\
& {} & {} \arrow[l, "h"'] & h \leftarrow \mathsf{Gen}(1^n) \\
x, r \leftarrow \{0,1\}^n & {} \arrow[r, "{c = (h(x), r, \langle r, x\rangle \oplus b)}"] & {} &
\end{tikzcd}$$

The fact that the scheme is computationaly binding follows immediately from the collision resistance of h: if the sender can find $(x, 0)$ and $(x', 1)$ that pass the receiver's verification, then $x \neq x'$ and $h(x) = h(x')$.

Arguing that the scheme is statistically-hiding is trickier. The reason is that $h(x)$ might reveal a lot of information on x. What helps us is that h is *shrinking*, and thus some information about x is hidden from the receiver. In particular, this means that x has positive min-entropy given $h(x)$. At this point we would like to apply the Leftover Hash Lemma (LHL) to show that for any b, the statistical distance between $(h(x), r, \langle r, x\rangle \oplus b)$ and $(h(x), r, u)$ is small. Unfortunately, the min-entropy level is insufficient to derive anything meaningful from the LHL and indeed the distance between these two distributions is a constant (rather than negligible as required).

To reduce the statistical distance, we increase the min-entropy via repetition. We modify the protocol so that the sender selects k values $\mathbf{x} = (x_1, \ldots, x_k) \leftarrow \{0,1\}^{n \cdot k}$ and $r \leftarrow \{0,1\}^{n \cdot k}$, and sends $\big(h(x_1), \ldots, h(x_k), r, \langle r, \mathbf{x} \rangle \oplus b\big)$ to the receiver. The min-entropy of $\mathbf{x}$, even given $h(x_1), \ldots, h(x_k)$ is now $\Omega(k)$, and the LHL now yields that the statistical distance between the two distributions $\big(h, h(x_1), \ldots, h(x_k), r, \langle r, \mathbf{x} \rangle \oplus 0\big)$ and $\big(h, h(x_1), \ldots, h(x_k), r, \langle r, \mathbf{x} \rangle \oplus 1\big)$ is roughly 2^{-k}. Setting k to be sufficiently large (e.g., $k = \mathsf{poly}(n)$ or even $k = \mathsf{poly}\log(n)$) we obtain that the scheme is statistically-hiding. Note that repetition also does not hurt binding: if the sender can find valid decommitments $(\mathbf{x} = (x_1 \ldots, x_k), 0)$ and $(\mathbf{x}' = (x'_1, \ldots, x'_k), 1)$ that pass the receiver's verification, then there must exist $i \in [k]$ with $x_i \neq x'_i$ and $h(x_i) = h(x'_i)$ (i.e., a collision).

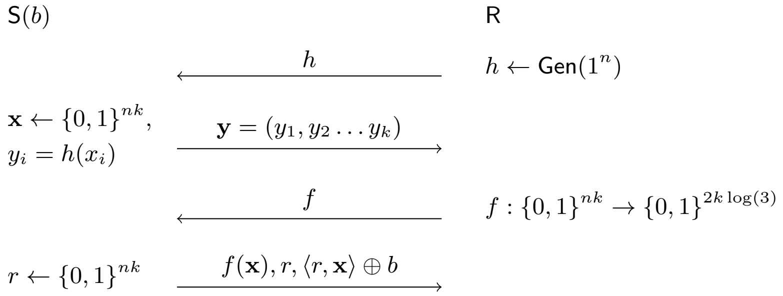
Handling MCRHs. For simplicity, let us focus on the case $t = 4$ (since it basically incorporates all the difficulty encountered when dealing with larger values of t). That is, we assume that $\mathcal{H} = \left\{h \colon \{0,1\}^n \to \{0,1\}^{n-s}\right\}$ is an (s,t)-MCRH with $s = 2$ and $t = 4$. Namely, it is hard to find 4 inputs that map to the same hash value for a random function from $\mathcal{H}$, even though such 4-way collisions exist. Note however that it might very well be easy to find 3 such colliding inputs. And indeed, the binding argument that we had before breaks: finding $x \neq x'$ with $h(x) = h(x)$ is no longer (necessarily) a difficult task.

The problem comes up because even after the sender 'commits' to $y_1 = h(x_1), \ldots, y_k = h(x_k)$, it is no longer forced to reveal $x_1, \ldots, x_k$. Intuitively, for every y_i, the sender might know 3 inputs that map to y_i, so, the sender is free to reveal any value in the Cartesian product of these triples. Concretely, let $\mathcal{S}_{y_i}$ be the set of inputs that h maps to y_i that the sender can find efficiently, and let $\mathcal{S}_{\mathbf{y}} = \mathcal{S}_{y_1} \times \cdots \times \mathcal{S}_{y_k}$. Since the sender can find at most 3 colliding inputs, it holds that $|\mathcal{S}_{y_i}| \leq 3$ for every i, and thus $|\mathcal{S}_{\mathbf{y}}| \leq 3^k$. To fix the binding argument, we want to force every efficient sender to able to reveal a *unique* $\mathbf{x} = (x_1, \ldots, x_k) \in \mathcal{S}_{\mathbf{y}}$.

A first attempt toward achieving the above goal is to try to use a pairwise-independent hash function f that is injective over $\mathcal{S}_{\mathbf{y}}$ with high probability. At a high level, the receiver will also specify to the sender a random function f from the pairwise independent hash function family. The sender in turn sends $f(\mathbf{x})$ as well as $(h(x_1), \ldots, h(x_k))$. The receiver adds a check to the verification step to ensure that f maps the decommited input sequence $(x'_1, \ldots, x'_k)$ to the value that was pre-specified.

In order for the function f to be injective on the set $\mathcal{S}_{\mathbf{y}}$, the birthday paradox tells us that the range of f must have size at least $|\mathcal{S}_{\mathbf{y}}|^2$ (roughly), which means at least 3^{2k}. Thus, to ensure that f is injective on $\mathcal{S}_{\mathbf{y}}$, we can use a pairwise-independent function $f \colon \{0,1\}^{nk} \to \{0,1\}^{2k \log(3)}$.

Unfortunately, this scheme is still not binding: f is promised (with high probability) to be injective for *fixed* sets of size 3^k, but the sender can choose $\mathbf{y}$ based on the value of f. Specifically, to choose $\mathbf{y}$ so that f is not injective over $\mathcal{S}_{\mathbf{y}}$. To fix the latter issue, we split the messages that the receiver sends into two rounds. In the first round the receiver sends h and receives $\mathbf{y} = \big(h(x_1), \ldots, h(x_k)\big)$ from the sender. Only then the receiver sends f and receives $z_1 = f(\mathbf{x})$. Now, the scheme is binding: since f is chosen *after* $\mathbf{y}$ is set, the pairwise-independence property guarantees that f will be injective over $\mathcal{S}_{\mathbf{y}}$ with high probability. Pictorially, the commit stage of the new scheme is as follows:


$\mathsf{S}(b)$
R
h
$h \leftarrow \mathsf{Gen}(1^n)$
$\mathbf{x} \leftarrow \{0,1\}^{nk}$,
$y_i = h(x_i)$
$\mathbf{y} = (y_1, y_2 \ldots y_k)$
f
$f : \{0,1\}^{nk} \to \{0,1\}^{2k\log(3)}$
$r \leftarrow \{0,1\}^{nk}$
$f(\mathbf{x}), r, \langle r, \mathbf{x} \rangle \oplus b$

But is this scheme statistically-hiding? Recall that previously, to argue hiding, we used the fact that the mapping $(x_1, \ldots, x_k) \mapsto (h(x_1), \ldots, h(x_k))$ is shrinking. In an analogous manner, here, we need the mapping $(x_1, \ldots, x_k) \mapsto \big(h(x_1), \ldots, h(x_k), f(\mathbf{x})\big)$ to be shrinking. However, the latter mapping maps strings of length $n \cdot k$ bits to strings of length $(n-2) \cdot k + 2\log(3) \cdot k$, which is obviously not shrinking.

One work-around is to simply assume that the given MCRH shrinks much more than we assumed so far. For example, to assume that $\mathcal{H}$ is $(4,4)$-MCRH (or more generally (s,t)-MCRH for $s \gg \log(t)$).[16] However, by adding one more round of interaction we can actually fix the protocol so that it gives statistically-hiding commitments even with tight shrinkage of $\log(t)$.

Overcoming the Birthday Paradox. To guarantee hiding, it seems that we cannot afford the range of f to be as large as $(3^k)^2$. Instead, we set its range size to 3^k (i.e., $f \colon \{0,1\}^{nk} \to \{0,1\}^{k\log(3)}$). Moreover, rather than choosing it from a pairwise independent hash function family, we shall one more use one that is *many-wise-independent*. Again, the important property that we use is that such functions are *load-balanced*[17] with high probability, z_1 — the value that the sender sends in the second round — has at most $\log(3^k) = k \cdot \log(3)$ pre-images

[16] We remark that our construction of MCRH based on $\mathsf{EA}_{\min,\max}$ (see Section 3) actually supports such large shrinkage.

[17] In a nutshell, the property that we are using is that if $N = 3^k$ balls are thrown into N bins, with high probability the maximal load in every bin will be at most $\log(N)$. It is well-known that hash functions that are $\log(N)$-wise independent also have this property. See Section 2.2 for details.

from $\mathcal{S}_{\mathbf{y}}$ under f (i.e., $|\{\mathbf{x} \in \mathcal{S}_{\mathbf{y}} \colon f(\mathbf{x}) = z_1\}| \leq k \cdot \log(3)$). We once more face the problem that the sender can reveal any of these inputs, but now their number is exponentially smaller — it is only $k \log(3)$ (as opposed to 3^k before). We can now choose a pairwise-independent $g \colon \{0,1\}^{nk} \to \{0,1\}^{2(\log(k)+\log\log(3))}$ that is injective over sets of size $k \cdot \log(3)$ (with high probability). For the same reasons that f was sent after h, the receiver sends g only after receiving $f(\mathbf{x})$.

Thus, our final protocol has three rounds (where each round is composed of one message for each of the two parties) and is as follows: In the first round, the receiver selects $h \leftarrow \mathcal{H}$ and sends it to the sender. The sender, trying to commit to a bit b, chooses $\mathbf{x} = (x_1, \dots, x_k) \leftarrow \{0,1\}^{nk}$ and sends $\mathbf{y} = (y_1 = h(x_1), \dots, y_k = h(x_k))$. In the second round, the receiver selects a many-wise-independent hash function $f \colon \{0,1\}^{nk} \to \{0,1\}^{k\log(3)}$ and sends it to the sender. The sender sends $z_1 = f(\mathbf{x})$ to the receiver. In the third and final round, the receiver selects a pairwise-independent hash function $g \colon \{0,1\}^{n \cdot k} \to \{0,1\}^{2(\log(k)+\log\log(3))}$ and sends it to the sender. The sender selects $r \leftarrow \{0,1\}^{nk}$, and sends $(z_2 = g(\mathbf{x}), r, \sigma = \langle r, \mathbf{x} \rangle \oplus b)$ to the receiver. The commitment is defined as $c = (h, \mathbf{y}, f, z_1, g, z_2, \sigma)$. To reveal, the sender sends $(\mathbf{x}, b)$ to the receiver, which verifies that $h(x_i) = y_i$ for every i, that $f(\mathbf{x}) = z_1$, $g(\mathbf{x}) = z_2$ and $\sigma = \langle r, \mathbf{x} \rangle \oplus b$. Pictorially, the commit stage is as follows:

$$\begin{array}{lcl}
\mathsf{S}(b) & & \mathsf{R} \\
 & \xleftarrow{\qquad h \qquad} & h \leftarrow \mathsf{Gen}(1^n) \\
\mathbf{x} \leftarrow \{0,1\}^{nk}, & \xrightarrow{\mathbf{y} = (y_1, y_2 \dots y_k)} & \\
y_i = h(x_i) & & \\
 & \xleftarrow{\qquad f \qquad} & f : \{0,1\}^{nk} \to \{0,1\}^{k\log(3)} \\
 & \xrightarrow{\qquad f(\mathbf{x}) \qquad} & \\
 & \xleftarrow{\qquad g \qquad} & g : \{0,1\}^{nk} \to \{0,1\}^{2(\log k + \log\log(3))} \\
r \leftarrow \{0,1\}^{nk} & \xrightarrow{g(\mathbf{x}), r, \langle r, \mathbf{x} \rangle \oplus b} &
\end{array}$$

Intuitively, the scheme is computationally binding since for any computationally bounded sender that committed to c, there is a unique $\mathbf{x}$ that passes the receiver's verification. As for hiding, we need the mapping $(x_1, \dots, x_k) \mapsto (h(x_1), \dots, h(x_k), f(\mathbf{x}), g(\mathbf{x}))$ to be shrinking. Observe that we are mapping $n \cdot k$ bits to $(n-2)k + \log(3)k + 2(\log(k) + \log\log(3))$ bits (where all logarithms are to the base 2). Choosing k to be sufficiently large (e.g., $k = \mathsf{poly}(n)$ certainly suffices) yields that the mapping is shrinking.

This completes the high level overview of the direct analysis of our construction of constant-round statistically hiding commitments. The formal proof, done via a reduction from the binding of the scheme to the MCRH property, requires more delicate care (and in particular handling certain probabilistic dependencies that arise in the reduction). See Section 4 for details.

1.3.2.2 Analysis via Inaccesible Entropy

Consider the jointly distributed random variables $(h(x), x)$, where h is chosen at random from a family of t-way collision resistant hash functions $\mathcal{H} = \left\{ h \colon \{0,1\}^n \to \{0,1\}^{n-\log(t)} \right\}$ and x is a uniform n-bit string. Since $h(x)$ is only $(n - \log(t))$ bits long, it can reveal only that amount of information about x. Thus, the entropy of x given $h(x)$ (and h) is at least $\log(t)$. In fact, a stronger property holds: the expected number of pre-images of $h(x)$, over the choice of x, is t. This implies that x given $h(x)$ has $\log(t)$ bits of (a weaker variant of) min-entropy.

While $h(x)$ has t pre-images (in expectation), no *efficient* strategy can find more than $t-1$ of them. Indeed, efficiently finding t such (distinct) pre-images directly violates the t-way collision resistance of h.

In terms of inaccessible entropy, the foregoing discussion establishes that $(h(x), x)$ is a 2-block inaccessible entropy generator where the second block (i.e., x) has real min-entropy $\log(t)$ and accessible max-entropy at most $\log(t-1)$. This block generator is not quite sufficient to get statistically-hiding commitment since the construction of [HRVW09, HV17] requires a larger gap between the entropies. This, however, is easily solved since taking many copies of the same generator increases the entropy gap. That is, the final 2-block generator is $\Big((h(x_1), \ldots, h(x_k)), (x_1, \ldots, x_k)\Big)$, for a suitable choice of k. The existence of constant-round statistically-hiding commitment now follows immediately from [HV17, Lemma 19].[18] The resulting protocol turns out to be essentially the same as that obtained by the direct analysis discussed above (and proved in Section 4).

1.4 Organization

In Section 2 we provide standard definitions and basic facts. In Section 3 we formally state the entropy approximation assumption and present our construction of MCRH based on this assumption. Lastly, In Section 4 we describe the construction of constant-round statistically-hiding commitments from MCRH.

As already mentioned, we defer the proof of the blackbox separation of MCRH from one-way permutations to the full version of this paper [BDRV17].

[18] The general construction of statistically-hiding commitments from inaccessible entropy generators is meant to handle a much more general case than the one needed in our setting. In particular, a major difficulty handled by [HRVW09, HV17] is when the generator has many blocks and it is not known in which one there is a gap between the real and accessible entropies.

2 Preliminaries

We use lowercase letters for values, uppercase for random variables, uppercase calligraphic letters (e.g., $\mathcal{U}$) to denote sets, boldface for vectors (e.g., $\mathbf{x}$), and uppercase sans-serif (e.g., A) for algorithms (i.e., Turing Machines). All logarithms considered here are in base two. We let poly denote the set of all polynomials. A function $\nu\colon \mathbb{N} \to [0,1]$ is *negligible*, denoted $\nu(n) = \mathsf{negl}(n)$, if $\nu(n) < 1/p(n)$ for every $p \in \mathsf{poly}$ and large enough n.

Given a random variable X, we write $x \leftarrow X$ to indicate that x is selected according to X. Similarly, given a finite set $\mathcal{S}$, we let $s \leftarrow \mathcal{S}$ denote that s is selected according to the uniform distribution on $\mathcal{S}$. We adopt the convention that when the same random variable occurs several times in an expression, all occurrences refer to a single sample. For example, $\Pr[f(X) = X]$ is defined to be the probability that when $x \leftarrow X$, we have $f(x) = x$. We write U_n to denote the random variable distributed uniformly over $\{0,1\}^n$. The support of a distribution D over a finite set $\mathcal{U}$, denoted $\mathrm{Supp}(D)$, is defined as $\{u \in \mathcal{U} : D(u) > 0\}$. The *statistical distance* of two distributions P and Q over a finite set $\mathcal{U}$, denoted as $\mathrm{SD}(P,Q)$, is defined as $\max_{\mathcal{S}\subseteq\mathcal{U}} |P(\mathcal{S}) - Q(\mathcal{S})| = \frac{1}{2}\sum_{u\in\mathcal{U}} |P(u) - Q(u)|$.

2.1 Many-wise Independent Hashing

Many-wise independent hash functions are used extensively in complexity theory and cryptography.

Definition 2.1 (ℓ-wise Independent Hash Functions). *For $\ell \in \mathbb{N}$, a family of functions $\mathcal{F} = \{f : \{0,1\}^n \to \{0,1\}^m\}$ is* ℓ-wise independent *if for every distinct $x_1, x_2, \ldots, x_\ell \in \{0,1\}^n$ and every $y_1, y_2, \ldots, y_\ell \in \{0,1\}^m$, it holds that*

$$\Pr_{f\leftarrow\mathcal{F}}[f(x_1) = y_1 \wedge f(x_2) = y_2 \wedge \cdots \wedge f(x_\ell) = y_\ell] = \frac{1}{M^\ell}.$$

Note that if $\mathcal{H}$ is k-wise independent for $k \geq 2$, it is also universal. The existence of efficient many-wise hash function families is well known.

Fact 2.2 (c.f. [Vad12, Corollary 3.34]) *For every $n, m, \ell \in \mathbb{N}$, there exists a family of ℓ-wise independent hash functions $\mathcal{F}^{(\ell)}_{n,m} = \{f\colon \{0,1\}^n \to \{0,1\}^m\}$ where a random function from $\mathcal{F}^{(\ell)}_{n,m}$ can be selected using $\ell \cdot \max(m,n)$ bits, and given a description of $f \in \mathcal{F}^{(\ell)}_{n.m}$ and $x \in \{0,1\}^n$, the value $f(x)$ can be evaluated in time $\mathsf{poly}(n,m,\ell)$.*

Whenever we only need pairwise independent hash function $\mathcal{F}^{(2)}_{n,m}$, we remove the two from the superscript and simply write $\mathcal{F}_{n,m}$.

2.2 Load Balancing

The theory of load balancing deals with allocating elements into bins, such that no bin has too many elements. If the allocation is done at random, it can be shown that with high probability the max load (i.e., the number of elements in the largest bin) is not large. In fact, allocating via many-wise independent hash function also suffices.

Fact 2.3 (Folklore (see, e.g., [CRSW13])) *Let $n, m, \ell \in \mathbb{N}$ with $\ell \geq 2e$ (where e is the base of the natural logarithm) and let $\mathcal{F}_{n,m}^{(\ell)}$ be an ℓ-wise independent hash function family. Then, for every set $\mathcal{S} \subseteq \{0,1\}^n$ with $|\mathcal{S}| \leq 2^m$ it holds that:*

$$\Pr_{f \leftarrow \mathcal{F}_{n,m}^{(\ell)}} \left[\exists y \in \{0,1\}^m \textit{ such that } \left|f^{-1}(y) \cap \mathcal{S}\right| \geq \ell\right] \leq 2^{m-\ell},$$

where $f^{-1}(y) = \{x \in \{0,1\}^n \colon f(x) = y\}$.

Proof. Fix $y \in \{0,1\}^m$. It holds that

$$\begin{aligned}
&\Pr_{f \leftarrow \mathcal{F}_{n,m}^{(\ell)}} \left[\, \left|f^{-1}(y) \cap \mathcal{S}\right| \geq \ell\right] \\
&\qquad \leq \Pr_{f \leftarrow \mathcal{F}_{n,m}^{(\ell)}} \left[\exists \text{ distinct } x_1, \ldots, x_\ell \in \mathcal{S} \colon f(x_1) = y \wedge \cdots \wedge f(x_\ell) = y\right] \\
&\qquad \leq \sum_{\text{distinct } x_1, \ldots, x_\ell \in \mathcal{S}} \Pr_{f \leftarrow \mathcal{F}_{n,m}^{\ell}} \left[f(x_1) = y \wedge \cdots \wedge f(x_\ell) = y\right] \\
&\qquad \leq \binom{2^m}{\ell} \cdot \left(\frac{1}{2^m}\right)^{\ell} \\
&\qquad \leq \left(\frac{e \cdot 2^m}{\ell}\right)^{\ell} \cdot \left(\frac{1}{2^m}\right)^{\ell} \\
&\qquad \leq 2^{-\ell},
\end{aligned}$$

where the second inequality is by a union bound, the third inequality follows from the ℓ-wise independence of $\mathcal{F}_{n,m}^{(\ell)}$, the fourth inequality is by a standard bound on binomial coefficients, and the last inequality follows by our assumption that $\ell \geq 2e$.

Fact 2.3 follows from a union bound over all values of $y \in \{0,1\}^m$. □

Remark 2.4 (More Efficient Hash Functions). We remark that more efficient constructions of hash functions guaranteeing the same load balancing performance as in Fact 2.3 are known in the literature.

Specifically, focusing on the setting of $\ell = O(m)$, Fact 2.3 gives a load balancing guarantee for functions whose description size (i.e., key length) is $\Omega(m^2)$ bits. In contrast, a recent result of Celis et al. [CRSW13] constructs such functions that require only $\tilde{O}(m)$ key size. Furthermore, a follow up work of Meka et al. [MRRR14] improves the *evaluation time* of the [CRSW13] hash function to be only poly-logarithmic in m (in the word RAM model).

However, since our focus is not on concrete efficiency, we ignore these optimizations throughout this work.

3 Constructing MCRH Families

In this section, we present a construction of a Multi-Collision Resistant Hash family (MCRH) based on the hardness of estimating certain notions of entropy of a distribution, given an explicit description of the distribution (i.e., a circuit that generates it). We define and discuss this problem in Section 3.1, and present the construction of MCRH in Section 3.2.

3.1 Entropy Approximation

In order to discuss the problem central to our construction, we first recall some standard notions of entropy.

Definition 3.1. *For a random variable X, we define the following notions of entropy:*

- **Min-entropy:** $\mathrm{H}_{\min}(X) = \min_{x \in \mathrm{Supp}(X)} \log\left(\frac{1}{\Pr[X=x]}\right)$.
- **Max-entropy:** $\mathrm{H}_{\max}(X) = \log\left(|\mathrm{Supp}(X)|\right)$.
- **Shannon entropy:** $\mathrm{H}_{\mathrm{Shannon}}(X) = \mathrm{E}_{x \leftarrow X}\left[\log\left(\frac{1}{\Pr[X=x]}\right)\right]$.

For any random variable, these entropies are related as described below. These relations ensure that the problems we describe later are well-defined.

Fact 3.2 *For a random variable X supported over $\{0,1\}^m$,*

$$0 \leq \mathrm{H}_{\min}(X) \leq \mathrm{H}_{\mathrm{Shannon}}(X) \leq \mathrm{H}_{\max}(X) \leq m.$$

Given a circuit $C : \{0,1\}^n \to \{0,1\}^m$, we overload C to also denote the random variable induced by evaluating C on a uniformly random input from $\{0,1\}^n$. With this notation, the Entropy Approximation problem is defined as below.

Definition 3.3 (Min-Max Entropy Approximation). *Let $g = g(n) \in \mathbb{R}$ be a function such that $0 < g(n) < n$. The* min-max Entropy Approximation *problem with* gap g, *denoted $\mathsf{EA}^{(g)}_{\mathrm{min,max}}$, is a promise problem* (YES, NO) *for* $\mathrm{YES} = \{\mathrm{YES}_n\}_{n \in \mathbb{N}}$ *and* $\mathrm{NO} = \{\mathrm{NO}_n\}_{n \in \mathbb{N}}$, *where we define*

$$\begin{aligned} \mathrm{YES}_n &= \{(1^n, C_n, k) \ : \ \mathrm{H}_{\min}(C_n) \geq k\}, \text{ and} \\ \mathrm{NO}_n &= \{(1^n, C_n, k) \ : \ \mathrm{H}_{\max}(C_n) \leq k - g(n)\}, \end{aligned}$$

and where in both cases C_n is a circuit that takes n bits of input, and $k \in \{0, \ldots, n\}$.

We also define $\mathsf{EA}_{\mathrm{min,max}} = \mathsf{EA}^{(1)}_{\mathrm{min,max}}$. That is, when we omit the gap g we simply mean that $g = 1$.

The *Shannon* Entropy Approximation problem (where $\mathrm{H}_{\min}$ and $\mathrm{H}_{\max}$ above are replaced with $\mathrm{H}_{\mathsf{Shannon}}$), with constant gap, was shown by Goldreich et al. [GSV99] to be complete for the class NISZK (promise problems with non-interactive statistical zero knowledge proof systems). For a discussion of generalizations of Entropy Approximation to other notions of entropy, and other related problems, see [DGRV11].

3.1.1 The Assumption: Average-Case Hardness of Entropy Approximation.

Our construction of MCRH is based on the average-case hardness of the Entropy Approximation problem $\mathsf{EA}_{\min,\max}$ defined above (i.e., with gap 1). We use the following definition of average-case hardness of promise problems.

Definition 3.4 (Average-case Hardness). *We say that a promise problem* $\Pi = (\mathrm{YES}, \mathrm{NO})$*, where* $\mathrm{YES} = \{\mathrm{YES}_n\}_{n\in\mathbb{N}}$ *and* $\mathrm{NO} = \{\mathrm{NO}_n\}_{n\in\mathbb{N}}$*, is* average-case hard *if there is a probabilistic algorithm* S *such that* $\mathsf{S}(1^n)$ *outputs samples from* $(\mathrm{YES}_n \cup \mathrm{NO}_n)$*, and for every family of polynomial-sized circuits* $\mathsf{A} = (\mathsf{A}_n)_{n\in\mathbb{N}}$*,*

$$\Pr_{x\leftarrow\mathsf{S}(1^n)}[\mathsf{A}_n(x) = \Pi(x)] \le \frac{1}{2} + \mathsf{negl}(n),$$

where $\Pi(x) = 1$ *if* $x \in \mathrm{YES}$ *and* $\Pi(x) = 0$ *if* $x \in \mathrm{NO}$*. We call* S *a* hard-instance sampler *for* Π*. The quantity* $(\Pr_{x\leftarrow\mathsf{S}(1^n)}[\mathsf{A}_n(x) = \Pi(x)] - 1/2)$ *is referred to as the* advantage *the algorithm* A *has in deciding* Π *with respect to the sampler* S*.*

In our construction and proofs, it will be convenient for us to work with the problem $\mathsf{EA}_{\min,\max}^{(\lfloor\sqrt{n}\rfloor)}$ rather than $\mathsf{EA}_{\min,\max} = \mathsf{EA}_{\min,\max}^{(1)}$. At first glance $\mathsf{EA}_{\min,\max}^{(\lfloor\sqrt{n}\rfloor)}$ seems to be an easier problem because the gap here is $\lfloor\sqrt{n}\rfloor$, which is much larger. The following simple proposition shows that these two problems are in fact equivalent (even in their average-case complexity). The key idea here is repetition: given a circuit C, we can construct a new circuit C' that outputs C evaluated on independent inputs with a larger gap.

Proposition 3.5. $\mathsf{EA}_{\min,\max}^{(\lfloor\sqrt{n}\rfloor)}$ *is average-case hard if and only if* $\mathsf{EA}_{\min,\max}^{(1)}$ *is average-case hard.*

Proof Sketch. Note that any YES instance of $\mathsf{EA}_{\min,\max}^{(\lfloor\sqrt{n}\rfloor)}$ is itself a YES instance of $\mathsf{EA}_{\min,\max}^{(1)}$, and the same holds for NO instances. So the average-case hardness of $\mathsf{EA}_{\min,\max}^{(\lfloor\sqrt{n}\rfloor)}$ immediately implies that of $\mathsf{EA}_{\min,\max}^{(1)}$, with the same hard-instance sampler. In order to show the implication in the other direction, we show how to use a hard-instance sampler for $\mathsf{EA}_{\min,\max}^{(1)}$ to construct a hard-instance sampler S' for $\mathsf{EA}_{\min,\max}^{(\lfloor\sqrt{n}\rfloor)}$.

$\underline{\mathsf{S}' \text{ on input } (1^n)}$:

The Construction of MCRH

Let S be a hard-instance sampler for $\mathsf{EA}^{(\lfloor\sqrt{n}\rfloor)}_{\mathrm{min,max}}$.

$\mathsf{Gen}(1^n)$:

1. Sample $(1^n, C_n, k) \leftarrow \mathsf{S}(1^n)$, where C_n maps $\{0,1\}^n \to \{0,1\}^{n'}$.
2. Sample[a] $f \leftarrow \mathcal{F}^{(3n)}_{n,(n-k)}$ and $g \leftarrow \mathcal{F}^{(2n)}_{(n'+n-k),(n-\lfloor\sqrt{n}\rfloor)}$.
3. Output the circuit that computes the function $h_{C_n,f,g} : \{0,1\}^n \to \{0,1\}^{n-\lfloor\sqrt{n}\rfloor}$ that is defined as follows:
$$h_{C_n,f,g}(x) \coloneqq g\big(C_n(x), f(x)\big).$$

[a] Recall that $\mathcal{F}^{(\ell)}_{n,m} = \{f : \{0,1\}^n \to \{0,1\}^m\}$ is a family of ℓ-wise independent hash functions.

Fig. 1: Construction of MCRH from Entropy Approximation.

1. Let $\ell = \lfloor\sqrt{n}\rfloor$. S' samples $(1^\ell, C_\ell, k) \leftarrow \mathsf{S}(1^\ell)$.
2. Let $\widehat{C}_n$ be the following circuit that takes an n-bit input x. It breaks x into $\ell+1$ disjoint blocks $x_1, \ldots, x_{\ell+1}$, where $x_1, \ldots, x_\ell$ are of size ℓ, and $x_{\ell+1}$ is whatever remains. It ignores $x_{\ell+1}$, runs a copy of C_ℓ on each of the other x_i's, and outputs a concatenation of all the outputs.
3. S' outputs $(1^n, \widehat{C}_n, k\cdot\ell)$.

As $\widehat{C}_n$ is the ℓ-fold repetition of C_ℓ, its max and min entropies are ℓ times the respective entropies of C_ℓ. So if C_ℓ had min-entropy at least k, then $\widehat{C}_n$ has min-entropy at least $k\cdot\ell$, and if C_ℓ had max-entropy at most $(k-1)$, then $\widehat{C}_n$ has max-entropy at most $(k-1)\cdot\ell = k\cdot\ell-\ell$, where $\ell = \lfloor\sqrt{n}\rfloor$. The proposition follows. □

3.2 The Construction

Our construction of a Multi-Collision Resistant Hash (MCRH) family is presented in Figure 1. We now prove that the construction is secure under our average-case hardness assumption.

Theorem 3.6. *If $\mathsf{EA}^{(\lfloor\sqrt{n}\rfloor)}_{\mathrm{min,max}}$ is average-case hard, then the construction in Figure 1 is an (s,t)-MCRH, where $s = \lfloor\sqrt{n}\rfloor$ and $t = 6n^2$.*

The above theorem, along with Proposition 3.5, now implies the following.

Corollary 3.7. *If $\mathsf{EA}_{\mathrm{min,max}}$ is average-case hard, then there exists an (s,t)-MCRH, where $s = \lfloor\sqrt{n}\rfloor$ and $t = 6n^2$.*

Note that above, the shrinkage being $\lfloor\sqrt{n}\rfloor$ guarantees that there exist $2^{\lfloor\sqrt{n}\rfloor}$-way collisions. But the construction is such that it is not possible to find even a $6n^2$-way collision, (which is sub-exponentially smaller). This is significant because, unlike in the case of standard collision-resistant hash functions (i.e., in which it is hard to find a pair of collisions), shrinkage in MCRHs cannot be easily amplified by composition while maintaining the same amount of collision-resistance (see Remark 1.2).

The rest of this section is dedicated to proving Theorem 3.6.

Proof of Theorem 3.6. Let Gen denote the algorithm described in Figure 1, and S be the hard-instance sampler used there. Fact 2.2, along with the fact that S runs in polynomial-time ensures that Gen runs in polynomial-time as well. The shrinkage requirement of an MCRH is satisfied because here the shrinkage is $s(n) = \lfloor\sqrt{n}\rfloor$. To demonstrate multi-collision resistance, we show how to use an adversary that finds $6n^2$ collisions in hash functions sampled by Gen to break the average-case hardness of $\mathsf{EA}^{(\lfloor\sqrt{n}\rfloor)}_{\mathrm{min,max}}$. For the rest of the proof, to avoid cluttering up notations, we will denote the problem $\mathsf{EA}^{(\lfloor\sqrt{n}\rfloor)}_{\mathrm{min,max}}$ by just EA.

We begin with an informal discussion of the proof. We first prove that large sets of collisions that exist in a hash function output by Gen have different properties depending on whether the instance that was sampled in step 1 of Gen was a YES or NO instance of EA. Specifically, notice that the hash functions that are output by Gen have the form $h_{C_n,f,g}(x) = g(C_n(x), f(x))$; we show that, except with negligible probability:

- In functions $h_{C_n,f,g}$ generated from $(1^n, C_n, k) \in \mathrm{YES}$, with high probability, there do not exist $3n$ distinct inputs $x_1, \ldots, x_{3n}$ such that they all have the same value of $(C_n(x_i), f(x_i))$.
- In functions $h_{C_n,f,g}$ generated from $(1^n, C_n, k) \in \mathrm{NO}$, with high probability, there do not exist $2n$ distinct inputs $x_1, \ldots, x_{2n}$ such that they all have distinct values of $(C_n(x_i), f(x_i))$, but all have the same value $g(C_n(x_i), f(x_i))$.

Note that in any set of $6n^2$ collisions for $h_{C_n,f,g}$, there has to be either a set of $3n$ collisions for (C_n, f) or a set of $2n$ collisions for g, and so at least one of the conclusions in the above two statements is violated.

A candidate average-case solver for EA, when given an instance $(1^n, C_n, k)$, runs steps 2 and 3 of the algorithm Gen from Figure 1 with this C_n and k. It then runs the collision-finding adversary on the hash function $h_{C_n,f,g}$ that is thus produced. If the adversary does not return $6n^2$ collisions, it outputs a uniformly random answer. But if these many collisions are returned, it checks which of the conclusions above is violated, and thus knows whether it started with a YES or NO instance. So whenever the adversary succeeds in finding collisions, the distinguisher can decide EA correctly with overwhelming probability. As long as the collision-finding adversary succeeds with non-negligible probability, then the distinguisher also has non-negligible advantage, contradicting the average-case hardness of EA.

We now state and prove the above claims about the properties of sets of collisions, then formally write down the adversary outlined above and prove that it breaks the average case hardness of EA.

The first claim is that for hash functions $h_{C_n,f,g}$ generated according to Gen using a YES instance, there is no set of $3n$ distinct x_i's that all have the same value for $C_n(x_i)$ and $f(x_i)$, except with negligible probability.

Claim 3.7.1 *Let* $(1^n, C_n, k)$ *be a* YES *instance of* EA. *Then,*

$$\Pr_{f \leftarrow \mathcal{F}^{(3n)}_{n,(n-k)}} \left[\exists y, y_1 \in \{0,1\}^* : \left|C_n^{-1}(y) \cap f^{-1}(y_1)\right| \geq 3n\right] \leq \frac{1}{2^n}.$$

Intuitively, the reason this should be true is that when C_n comes from a YES instance, it has high min-entropy. This means that for any y, the set $C_n^{-1}(y)$ will be quite small. The function f can now be thought of as partitioning each set $C_n^{-1}(y)$ into several parts, none of which will be too large because of the load-balancing properties of many-wise independent hash functions.

Proof. The above probability can be bounded using the union bound as follows:

$$\begin{aligned}\Pr_f&\left[\exists y, y_1 : \left|C_n^{-1}(y) \cap f^{-1}(y_1)\right| \geq 3n\right] \\ &\leq \sum_{y \in \mathrm{Im}(C_n)} \Pr_f\left[\exists y_1 : \left|C_n^{-1}(y) \cap f^{-1}(y_1)\right| \geq 3n\right]. \qquad (1)\end{aligned}$$

The fact that $(1^n, C_n, k)$ is a YES instance of EA means that $\mathrm{H}_{\min}(C_n) \geq k$. The definition of min-entropy now implies that for any $y \in \mathrm{Im}(C_n)$:

$$\log\left(\frac{1}{\Pr_{x \leftarrow \{0,1\}^n}[C_n(x) = y]}\right) \geq k,$$

which in turn means that $\left|C_n^{-1}(y)\right| \leq 2^{n-k}$. Fact 2.3 (about the load-balancing properties of $\mathcal{F}^{(3n)}_{n,(n-k)}$) now implies that for any $y \in \mathrm{Im}(C_n)$:

$$\Pr_f\left[\exists y_1 : \left|C_n^{-1}(y) \cap f^{-1}(y_1)\right| \geq 3n\right] \leq \frac{2^{n-k}}{2^{3n}} \leq \frac{1}{2^{2n}}. \qquad (2)$$

Combining Eqs. (1) and (2), and noting that the image of C_n has at most 2^n elements, we get the desired bound:

$$\Pr_f\left[\exists y, y_1 : \left|C_n^{-1}(y) \cap f^{-1}(y_1)\right| \geq 3n\right] \leq 2^n \cdot \frac{1}{2^{2n}} \leq \frac{1}{2^n}.$$

□

The next claim is that for hash functions $h_{C_n,f,g}$ generated according to Gen using a NO instance, there is no set of $2n$ values of x_i that all have distinct values of $(C_n(x_i), f(x_i))$, but the same value $g(C_n(x_i), f(x_i))$, except with negligible probability.

Claim 3.7.2 *Let $(1^n, C_n, k)$ be a* NO *instance of* EA. *Then,*

$$\Pr_{\substack{f \leftarrow \mathcal{F}^{(3n)}_{n,(n-k)} \\ g \leftarrow \mathcal{F}^{(2n)}_{(n'+n-k),(n-\lfloor\sqrt{n}\rfloor)}}} \left[\exists x_1, \ldots, x_{2n} : \begin{array}{c} \textit{For all } i \neq j, \\ \big(C_n(x_i), f(x_i)\big) \neq \big(C_n(x_j), f(x_j)\big) \\ \textit{and} \\ g\big(C_n(x_i), f(x_i)\big) = g\big(C_n(x_j), f(x_j)\big) \end{array} \right] \leq \frac{1}{2^n}.$$

Proof. The fact that $(1^n, C_n, k)$ is a NO instance of EA means that $\mathrm{H}_{\max}(C_n) \leq k - \lfloor\sqrt{n}\rfloor$; that is, C_n has a small range: $|\mathrm{Im}(C_n)| \leq 2^{k-\lfloor\sqrt{n}\rfloor}$.

For any $f \in \mathcal{F}^{(3n)}_{n,(n-k)}$, which is what is sampled by Gen when this instance is used, the range of f is a subset of $\{0,1\}^{n-k}$. This implies that even together, C_n and f have a range whose size is bounded as:

$$|\mathrm{Im}(C_n, f)| \leq 2^{k-\lfloor\sqrt{n}\rfloor} \cdot 2^{n-k} = 2^{n-\lfloor\sqrt{n}\rfloor},$$

where (C_n, f) denotes the function that is the concatenation of C_n and f.

For there to exist a set of $2n$ inputs x_i that all have distinct values for $(C_n(x_i), f(x_i))$ but the same value for $g(C_n(x_i), f(x_i))$, there has to be a y that has more than $2n$ inverses under g that are all in the image of (C_n, f). As g comes from $\mathcal{F}^{(2n)}_{(n'+n-k),(n-\lfloor\sqrt{n}\rfloor)}$, we can use Fact 2.3 along with the above bound on the size of the image of (C_n, f) to bound the probability that such a y exists as follows:

$$\Pr_g\big[\exists y : \big|g^{-1}(y) \cap \mathrm{Im}(C_n, f)\big| \geq 2n\big] \leq \frac{2^{n-\lfloor\sqrt{n}\rfloor}}{2^{2n}} \leq \frac{1}{2^n}.$$

□

Let $\mathsf{A} = (\mathsf{A}_n)_{n\in\mathbb{N}}$ be a polynomial-size family of circuits that given a hash function output by $\mathsf{Gen}(1^n)$ finds a $6n^2$-way collision in it with non-negligible probability. The candidate circuit family $\mathsf{A}' = (\mathsf{A}'_n)_{n\in\mathbb{N}}$ for solving EA on average is described below.

A'_n on input $(1^n, C_n, k)$:

1. Run steps 2 and 3 of the algorithm Gen in Figure 1 with $(1^n, C_n, k)$ in place of the instance sampled from S there. This results in the description of a hash function $h_{C_n,f,g}$.
2. Run $\mathsf{A}_n(h_{C_n,f,g})$ to get a set of purported collisions $\mathcal{S}$.
3. If $\mathcal{S}$ does not actually contain $6n^2$ collisions under $h_{C_n,f,g}$, output a random bit.
4. If $\mathcal{S}$ contains $3n$ distinct x_i's such that they all have the same value of $(C_n(x_i), f(x_i))$, output 0.
5. If $\mathcal{S}$ contains $2n$ distinct x_i's such that they all have distinct values of $(C_n(x_i), f(x_i))$ but the same value $g(C_n(x_i), f(x_i))$, output 1.

. . . .

The following claim now states that any collision-finding adversary for the MCRH constructed can be used to break the average-case hardness of EA, thus completing the proof.

Claim 3.7.3 *If* A *finds* $6n^2$ *collisions in hash functions output by* $\mathsf{Gen}(1^n)$ *with non-negligible probability, then* A' *has non-negligible advantage in deciding* EA *with respect to the hard-instance sampler* S *used in* Gen*.*

Proof. On input $(1^n, C_n, k)$, the adversary A'_n computes $h_{C_n,f,g}$ and runs A_n on it. If A_n does not find $6n^2$ collisions for $h_{C_n,f,g}$, then A'_n guesses at random and is correct in its output with probability $1/2$. If A_n does find $6n^2$ collisions, then A'_n is correct whenever one of the following is true:

1. $(1^n, C_n, k)$ is a YES instance and there is no set of $3n$ collisions for (C_n, f).
2. $(1^n, C_n, k)$ is a NO instance and there is no set of $2n$ collisions for g in the image of (C_n, f).

Note that inputs to A'_n are drawn from $\mathsf{S}(1^n)$, and so the distribution over $h_{C_n,f,g}$ produced by A'_n is the same as that produced by $\mathsf{Gen}(1^n)$ itself. With such samples, let E_1 denote the event of (C_n, f) having a set of $3n$ collisions from $\mathcal{S}$ (the set output by A_n), and let E_2 denote the event of g having a set of $2n$ collisions in the image of (C_n, f) from $\mathcal{S}$. Also, let E_Y denote the event of the input to A'_n being a YES instance, E_N that of it being a NO instance, and E_{A} the event that $\mathcal{S}$ contains at least $6n^2$ collisions.

Following the statements above, the probability that A'_n is *wrong* in deciding EA with respect to $(1^n, C_n, k) \leftarrow \mathsf{S}(1^n)$ can be upper-bounded as:

$$\begin{aligned}
&\Pr\left[\mathsf{A}'_n(1^n, C_n, k) \text{ is wrong}\right] \\
&\qquad = \Pr\left[(\neg E_{\mathsf{A}}) \wedge (\mathsf{A}'_n \text{ is wrong})\right] + \Pr\left[E_{\mathsf{A}} \wedge (\mathsf{A}'_n \text{ is wrong})\right] \\
&\qquad \leq \Pr[\neg E_{\mathsf{A}}] \cdot \frac{1}{2} + \Pr[(E_Y \wedge E_1) \vee (E_N \wedge E_2)].
\end{aligned}$$

The first term comes from the fact that if A_n doesn't find enough collisions, A'_n guesses at random. The second term comes from the fact that if both $(E_Y \wedge E_1)$ and $(E_N \wedge E_2)$ are false and E_{A} is true, then since at least one of E_Y and E_N is always true, one of $(E_Y \wedge \neg E_1)$ and $(E_N \wedge \neg E_2)$ will also be true, either of which would ensure that A'_n is correct, as noted earlier.

We now bound the second term above, starting as follows:

$$\begin{aligned}
\Pr[(E_Y \wedge E_1) \vee (E_N \wedge E_2)] &\leq \Pr[(E_Y \wedge E_1)] + \Pr[(E_N \wedge E_2)] \\
&= \Pr[E_Y]\Pr[E_1|E_Y] + \Pr[E_N]\Pr[E_2|E_N] \\
&\leq \Pr[E_Y] \cdot \mathsf{negl}(n) + \Pr[E_N] \cdot \mathsf{negl}(n) \\
&= \mathsf{negl}(n),
\end{aligned}$$

where the first inequality follows from the union bound and the last inequality follows from Claims 3.7.1 and 3.7.2.

Putting this back in the earlier expression,

$$\begin{aligned}\Pr\left[\mathsf{A}'_n(1^n, C_n, k) \text{ is wrong}\right] &\leq \Pr[\neg E_{\mathsf{A}}] \cdot \frac{1}{2} + \mathsf{negl}(n)\\ &= \frac{1}{2} - \frac{\Pr[E_{\mathsf{A}}]}{2} + \mathsf{negl}(n).\end{aligned}$$

In other words,

$$\Pr\left[\mathsf{A}'_n(1^n, C_n, k) \text{ is correct}\right] \geq \frac{1}{2} + \frac{\Pr[E_{\mathsf{A}}]}{2} - \mathsf{negl}(n).$$

So if A succeeds with non-negligible probability in finding $6n^2$ collisions, then A' had non-negligible advantage in deciding EA over S. □

This concludes the proof of Theorem 3.6. □

4 Constant-Round Statistically-Hiding Commitments

In this section we show that multi-collision-resistant hash functions imply the existence of *constant-round* statistically-hiding commitments. Here we follow the "direct route" discussed in the introduction (rather than the "inaccessible entropy route").

For simplicity, we focus on *bit* commitment schemes (in which messages are just single bits). As usual, full-fledged commitment schemes (for long messages) can be obtained by committing bit-by-bit.

Definition 4.1 (Bit Commitment Scheme). *A* bit commitment scheme *is an interactive protocol between two polynomial-time parties — the* sender S *and the* receiver R *— that satisfies the following properties.*

1. *The protocol proceeds in two stages: the* commit *stage and the* reveal *stage.*
2. *At the start of the commit stage both parties get a security parameter 1^n as a common input and the sender S also gets a private input $b \in \{0, 1\}$. At the end of the commit stage the parties have a shared output c, which is called the* commitment*, and the sender S has an additional private output d, which is called the* decommitment*.*
3. *In the reveal stage, the sender S sends (b, d) to the receiver R. The receiver R accepts or rejects based on c, d and b. If both parties follow the protocol, then the receiver R always accepts.*

In this section we focus on commitment schemes that are *statistically*-hiding and *computationally*-binding.

Definition 4.2 (Statistically Hiding Bit Commitment). *A bit commitment scheme* (S, R) *is* statistically-hiding *if for every cheating receiver* R^* *it holds that*

$$\mathrm{SD}((\mathsf{S}(0), \mathsf{R}^*)(1^n), (\mathsf{S}(1), \mathsf{R}^*)(1^n)) = \mathsf{negl}(n),$$

where $(\mathsf{S}(b), \mathsf{R}^*)(1^n)$ *denotes the transcript of the interaction between* R^* *and* $\mathsf{S}(b)$ *in the commit stage.*

Definition 4.3 (Computationally Binding Bit Commitment). *A bit commitment scheme* (S, R) *is said to be* computationally-binding *if for every family of polynomial-size circuits sender* $\mathsf{S}^* = (\mathsf{S}^*_n)_{n\in\mathbb{N}}$ *it holds that* S^* *wins in the following game with only with* $\mathsf{negl}(n)$ *probability:*

1. *The cheating sender* S^*_n *interacts with the honest receiver* $\mathsf{R}(1^n)$ *in the commit stage obtaining a commitment* c*.*
2. *Then,* S^*_n *outputs two pairs* $(0, d_0)$ *and* $(1, d_1)$*. The cheating sender* S^* *wins if the honest receiver* R *accepts both* $(c, 0, d_0)$ *and* $(c, 1, d_1)$*.*

We are now ready to state the main result of this section. A *round* of a commitment scheme is a pair of messages, the first sent from the receiver to the sender, and the second the other way.

Theorem 4.4 ($\mathsf{MCRH} \implies$ **Constant-Round Statistically-Hiding Commitments).** *Let* $t = t(n) \in \mathbb{N}$ *be a polynomial computable in* $\mathsf{poly}(n)$ *time. Assume that there exists a* (s, t)*-*MCRH *for* $s \geq \log(t)$*, then there exists a* three-round *statistically-hiding computationally-binding commitment scheme.*

As we already mentioned in Section 1, constructions of statistically-hiding computationally-binding commitment schemes are known assuming only the minimal assumption that one-way functions exist. Those constructions, however, have a polynomial number of rounds (and this is inherent for black-box constructions [HHRS15]). Theorem 4.4, on the other hand, yields a commitment scheme with only a constant (i.e., three) number of rounds.

Due to space limitation, we defer the complete proof of Theorem 4.4 to the full version of this paper [BDRV17].

Acknowledgments

We thank Vinod Vaikuntanathan for helpful discussions and for his support, and Oded Goldreich, Yuval Ishai and the anonymous reviewers for useful comments. We thank Nir Bitansky, Yael Kalai, Ilan Komargodski, Moni Naor, Omer Paneth and Eylon Yogev for helping us provide a good example of a t-way collision. We also thank Nir Bitansky and an anonymous reviewer for pointing out the connection to inaccessible entropy.

This research was supported in part by NSF Grants CNS-1413920 and CNS-1350619, and by the Defense Advanced Research Projects Agency (DARPA) and the U.S. Army Research Office under contracts W911NF-15-C-0226 and W911NF-15-C-0236. The third author was also partially supported by the SIMONS Investigator award agreement dated 6-5-12 and by the Cybersecurity and Privacy Institute at Northeastern University.

Bibliography

[ADM$^+$99] Noga Alon, Martin Dietzfelbinger, Peter Bro Miltersen, Erez Petrank, and Gábor Tardos. Linear hash functions. *J. ACM*, 46(5):667–683, 1999.

[AR16] Benny Applebaum and Pavel Raykov. On the relationship between statistical zero-knowledge and statistical randomized encodings. In *Annual Cryptology Conference*, pages 449–477. Springer, 2016.

[Bab16] László Babai. Graph isomorphism in quasipolynomial time [extended abstract]. In Daniel Wichs and Yishay Mansour, editors, *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 684–697. ACM, 2016.

[BDRV17] Itay Berman, Akshay Degwekar, Ron D. Rothblum, and Prashant Nalini Vasudevan. Multi collision resistant hash functions and their applications. *IACR Cryptology ePrint Archive*, 2017:489, 2017.

[BPK17] Nir Bitansky, Omer Paneth, and Yael Tauman Kalai. Multi-collision resistance: A paradigm for keyless hash functions. *Electronic Colloquium on Computational Complexity (ECCC)*, 24:99, 2017.

[BPVY00] Ernest F. Brickell, David Pointcheval, Serge Vaudenay, and Moti Yung. Design validations for discrete logarithm based signature schemes. In *Public Key Cryptography, Third International Workshop on Practice and Theory in Public Key Cryptography, PKC 2000, Melbourne, Victoria, Australia, January 18-20, 2000, Proceedings*, pages 276–292, 2000.

[CRSW13] L. Elisa Celis, Omer Reingold, Gil Segev, and Udi Wieder. Balls and bins: Smaller hash families and faster evaluation. *SIAM J. Comput.*, 42(3):1030–1050, 2013.

[DGRV11] Zeev Dvir, Dan Gutfreund, Guy N. Rothblum, and Salil P. Vadhan. On approximating the entropy of polynomial mappings. In *Innovations in Computer Science - ICS 2010, Tsinghua University, Beijing, China, January 7-9, 2011. Proceedings*, pages 460–475, 2011.

[DHRS07] Yan Zong Ding, Danny Harnik, Alon Rosen, and Ronen Shaltiel. Constant-round oblivious transfer in the bounded storage model. *J. Cryptology*, 20(2):165–202, 2007.

[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*, pages 711–720. ACM, 2006.

[DPP93] Ivan Damgård, Torben P. Pedersen, and Birgit Pfitzmann. On the existence of statistically hiding bit commitment schemes and fail-stop signatures. In *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*, pages 250–265, 1993.

[GG98] Oded Goldreich and Shafi Goldwasser. On the limits of non-approximability of lattice problems. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 1–9. ACM, 1998.

[GMR88] Shafi Goldwasser, Silvio Micali, and Ronald L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM J. Comput.*, 17(2):281–308, 1988.

[GS94] Marc Girault and Jacques Stern. On the length of cryptographic hash-values used in identification schemes. In *Advances in Cryptology - CRYPTO '94, 14th Annual International Cryptology Conference, Santa Barbara, California, USA, August 21-25, 1994, Proceedings*, pages 202–215, 1994.

[GSV99] Oded Goldreich, Amit Sahai, and Salil P. Vadhan. Can statistical zero knowledge be made non-interactive? or on the relationship of SZK and NISZK. In *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, pages 467–484, 1999.

[HHRS15] Iftach Haitner, Jonathan J Hoch, Omer Reingold, and Gil Segev. Finding collisions in interactive protocols—tight lower bounds on the round and communication complexities of statistically hiding commitments. *SIAM Journal on Computing*, 44(1):193–242, 2015.

[HM96] Shai Halevi and Silvio Micali. Practical and provably-secure commitment schemes from collision-free hashing. In *Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings*, pages 201–215, 1996.

[HNO$^+$09] Iftach Haitner, Minh-Huyen Nguyen, Shien Jin Ong, Omer Reingold, and Salil P. Vadhan. Statistically hiding commitments and statistical zero-knowledge arguments from any one-way function. *SIAM J. Comput.*, 39(3):1153–1218, 2009.

[HR04] Chun-Yuan Hsiao and Leonid Reyzin. Finding collisions on a public road, or do secure hash functions need secret coins? In *Advances in Cryptology - CRYPTO 2004, 24th Annual International Cryptology-Conference, Santa Barbara, California, USA, August 15-19, 2004, Proceedings*, pages 92–105, 2004.

[HRVW09] Iftach Haitner, Omer Reingold, Salil P. Vadhan, and Hoeteck Wee. Inaccessible entropy. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 611–620, 2009.

[HV17] Iftach Haitner and Salil P. Vadhan. The many entropies in one-way functions. In *Tutorials on the Foundations of Cryptography.*, pages 159–217. 2017.

[Jou04] Antoine Joux. Multicollisions in iterated hash functions. application to cascaded constructions. In *Advances in Cryptology - CRYPTO 2004, 24th Annual International CryptologyConference, Santa Bar-*

bara, California, USA, August 15-19, 2004, Proceedings, pages 306–316, 2004.

[Kil92] Joe Kilian. A note on efficient zero-knowledge proofs and arguments (extended abstract). In *Proceedings of the 24th Annual ACM Symposium on Theory of Computing, May 4-6, 1992, Victoria, British Columbia, Canada*, pages 723–732, 1992.

[KNY17a] Ilan Komargodski, Moni Naor, and Eylon Yogev. Collision resistant hashing for paranoids: Dealing with multiple collisions. *IACR Cryptology ePrint Archive*, 2017:486, 2017.

[KNY17b] Ilan Komargodski, Moni Naor, and Eylon Yogev. White-box vs. black-box complexity of search problems: Ramsey and graph property testing. *Electronic Colloquium on Computational Complexity (ECCC)*, 24:15, 2017.

[Mer89] Ralph C. Merkle. One way hash functions and DES. In *Advances in Cryptology - CRYPTO '89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings*, pages 428–446, 1989.

[Mic00] Silvio Micali. Computationally sound proofs. *SIAM J. Comput.*, 30(4):1253–1298, 2000.

[MRRR14] Raghu Meka, Omer Reingold, Guy N. Rothblum, and Ron D. Rothblum. Fast pseudorandomness for independence and load balancing - (extended abstract). In *Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I*, pages 859–870, 2014.

[NY89] Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing, May 14-17, 1989, Seattle, Washigton, USA*, pages 33–43, 1989.

[Ost91] Rafail Ostrovsky. One-way functions, hard on average problems, and statistical zero-knowledge proofs. In *Proceedings of the Sixth Annual Structure in Complexity Theory Conference, Chicago, Illinois, USA, June 30 - July 3, 1991*, pages 133–138, 1991.

[OV08] Shien Jin Ong and Salil P. Vadhan. An equivalence between zero knowledge and commitments. In *Theory of Cryptography, Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19-21, 2008.*, pages 482–500, 2008.

[PRS12] Krzysztof Pietrzak, Alon Rosen, and Gil Segev. Lossy functions do not amplify well. In *Theory of Cryptography - 9th Theory of Cryptography Conference, TCC 2012, Taormina, Sicily, Italy, March 19-21, 2012. Proceedings*, pages 458–475, 2012.

[PW11] Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. *SIAM J. Comput.*, 40(6):1803–1844, 2011.

[Rom90] John Rompel. One-way functions are necessary and sufficient for secure signatures. In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA*, pages 387–394, 1990.

[RS96] Ronald L. Rivest and Adi Shamir. Payword and micromint: Two simple micropayment schemes. In *Security Protocols, International Workshop, Cambridge, United Kingdom, April 10-12, 1996, Proceedings*, pages 69–87, 1996.

[RTV04] Omer Reingold, Luca Trevisan, and Salil P. Vadhan. Notions of reducibility between cryptographic primitives. In *Theory of Cryptography, First Theory of Cryptography Conference, TCC 2004, Cambridge, MA, USA, February 19-21, 2004, Proceedings*, pages 1–20, 2004.

[RV09] Guy N. Rothblum and Salil P. Vadhan, 2009. Unpublished Manuscript.

[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, International Conference on the Theory and Application of Cryptographic Techniques, Espoo, Finland, May 31 - June 4, 1998, Proceeding*, pages 334–345, 1998.

[Vad12] Salil P. Vadhan. Pseudorandomness. *Foundations and Trends in Theoretical Computer Science*, 7(1-3):1–336, 2012.