New Constructions of Collapsing Hashes

Mark Zhandry[1,2]
mzhandry@gmail.com

[1] NTT Research, Sunnyvale, USA
[2] Princeton University, Princeton, USA

Abstract. Collapsing is a post-quantum strengthening of collision resistance, needed to lift many classical results to the quantum setting. Unfortunately, the only existing standard-model proofs of collapsing hashes require LWE. We construct the first collapsing hashes from the quantum hardness of any one of the following problems:

- LPN in a variety of low noise or high-hardness regimes, essentially matching what is known for collision resistance from LPN.
- Finding cycles on exponentially-large expander graphs, such as those arising from isogenies on elliptic curves.
- The "optimal" hardness of finding collisions in *any* hash function.
- The *polynomial* hardness of finding collisions, assuming a certain plausible regularity condition on the hash.

As an immediate corollary, we obtain the first statistically hiding post-quantum commitments and post-quantum succinct arguments (of knowledge) under the same assumptions. Our results are obtained by a general theorem which shows how to construct a collapsing hash H' from a post-quantum collision-resistant hash function H, regardless of whether or not H itself is collapsing, assuming H satisfies a certain regularity condition we call "semi-regularity".

1 Introduction

Collision resistance is one of the most important cryptographic concepts, with numerous applications throughout cryptography. A collision resistant hash function $H : \{0,1\}^m \to \{0,1\}^n$ is one where $n < m$, thus guaranteeing that collisions exist in abundance, but where actually finding such collisions is computationally intractable. Collision resistance provably follows from most number-theoretic problems used in cryptography, and is one of the main design goals in constructions built from symmetric key tools, such as SHA2 or SHA3.

What happens when quantum computers enter the picture? For any application that required collision resistance classically, certainly a *minimal* condition is that it remains intractable for quantum algorithms to find a collision. We will call this notion a "post-quantum" collision resistant hash function (PQ-CRHF). Post-quantum security rules out constructions based on discrete logarithms or factoring due to Shor's algorithm [Sho94]. Surprisingly, however, even PQ-CRHFs are often *insufficient* for applications, as first demonstrated by Ambainis, Rosmanis, and Unruh [ARU14, Unr16b] with a counterexample. The issue usually stems

from rewinding, which is known to be problematic quantumly [VDG98, Wat06]. Examples include commitments and more generally interactive protocols.

To remedy the situation, Unruh [Unr16b] proposes a strengthening of collision resistance called *collapsing*. Very roughly, collapsing means that measuring the hash of a quantum superposition of messages is quantum computationally indistinguishable from measuring the message superposition itself, even though both operations are information-theoretically very different. Since its introduction, collapsing hashes have become recognized as the preferred notion of post-quantum security, being the appropriate post-quantum replacement for classical collision resistance whenever there is rewinding [CCY21, CMSZ21, LMS21], and sometimes even when rewinding is not present [AMRS20]. Unsurprisingly, collapsing is also a natural property beyond hash functions, being the right notion of post-quantum commitments [Unr16b] (whereas PQ computational binding is useless), identification protocols underlying post-quantum signatures [DFMS19, LZ19], and general argument systems [LMS21].

Given their importance to post-quantum security, it is crucial to understand how to construct collapsing hash functions. Unfortunately, there are essentially only two classes of constructions. The first are idealized model proofs [Unr16b, Unr17], where one proves collapsing relative to, say, a random oracle. The second are standard-model proofs [Unr16a, LZ19], where the only existing paradigm leverages lossy functions or closely related concepts, whose only known post-quantum instantiations require LWE (or equivalently, SIS by Regev's reduction [Reg05]).

On the other hand, the only hash functions which are provably PQ-CRHFs but *not* collapsing are contrived and require either complex oracles [Unr16b, AGKZ20] or un-tested conjectures [Zha19b][1]. Zhandry [Zha19b] even shows that such a separation between the notions could be used to build public key quantum money and stronger objects, which have been notoriously hard to build. In summary, neither of the following scenarios would contradict any long-standing conjectures:

- Collapsing is ubiquitous, and *every* non-relativized PQ-CRHF is collapsing.
- Collapsing is rare, and the *only* standard-model collapsing hash functions are those requiring LWE.

On Random Oracle-based Hashes. One may argue that we can simply conjecture that some hash function is collapsing, and then trivially "build" collapsing hashes from that function. In particular, random oracles are collapsing [Unr16b] and symmetric key hash functions such as SHA2 or SHA3 are often modeled as random oracles.

However, collapsing is an inherently quantum notion, which is potentially much harder to reason about than typical classically-defined notions such as collision resistance, pseudorandomness, etc. Indeed, the random oracle heuristic is based on extensive cryptanalytic studies of the hash functions with respect to classically-defined tasks. This is true even for works considering quantum attacks [HS21,

[1] [Zha19b] gives a proof relative to a novel computational assumption, but it has been cryptanalyzed [Rob21].

AMG+16], where the cryptanalysis goal is still classically-defined, such as finding collisions. Some works have proved the post-quantum *indifferentiability* of these functions [Zha19a, CHS19, Cza21]; while these are important for understanding security, they punt the cryptanalysis effort to the underlying round function, which again have largely been studied for their classical security.

Aside from idealized model justifications, we are not aware of *any* cryptanalysis effort on hash functions like SHA2 or SHA3 with regards to collapsing. Therefore, it seems plausible that the random oracle heuristic could hold on symmetric hash functions relative to classically-defined security properties, but fails for collapsing. For this reason, the current evidence for SHA2 or SHA3 being collapsing appears much weaker than evidence for their (post-quantum) collision resistance.

Our Results. In this work, we build a collapsing hash function H' from any PQ-CRHF H that satisfies a mild structural condition we call *semi-regularity*. Semi-regularity essentially means that no output has too many more pre-images than the "average" output. Note that H itself may be equivocal, and indeed the counter-example of [ARU14] is semi-regular. Yet when plugged into our construction, the resulting H' is collapsing. We then show the following:

- Hash functions based on expanders [TZ94, CLG09, FLLT21], or a variety of LPN settings [BLVW19, YZW+19] satisfy our regularity condition. In these cases, we thus achieve collapsing hashes under the same assumptions used to achieve post-quantum collision resistance.
- We do not know how to prove semi-regularity for symmetric hash function such as SHA2 or SHA3, but it is a natural property and it is reasonable to conjecture it holds for these functions. In particular, random oracles are semi-regular. Under this conjecture together with post-quantum collision resistance for SHA2 or SHA3, we obtain collapsing hashes. This is the first standard-model collapsing hash function from *classically defined* assumptions in Minicrypt; that is, they do not imply public key encryption.
- As an alternative approach, we show that H can be compiled into a collapsing hash function if it is *optimally* collision resistant, even if it is not semi-regular. Optimal collision resistance means that every polynomial-time algorithm can only find collisions with probability $\mathsf{poly}/|\mathsf{Range}|$. Note that the optimal generic classical and quantum [BHT97] collision-finding algorithms make T queries and succeed with probability $O(T^2)/|\mathsf{Range}|$ and $O(T^3)/|\mathsf{Range}|$, respectively. Symmetric hashes such as SHA2 or SHA3 are often designed with the goal of achieving optimal collision resistance, and so we obtain collapsing hashes under the assumed optimal collision resistance of either of these functions.

As immediate corollaries of our results, we obtain post-quantum statistically hiding commitments [Unr16b] and succinct arguments [CMSZ21] under any of the above assumptions. Our results show that semi-regularity is an important design consideration for constructing post-quantum hash functions.

1.1 Why PQ-CRHFs Are Not Enough

For completeness, we give a brief explanation of why rewinding is problematic with PQ-CRHFs. Consider the following game. An adversary sends a hash y to the challenger. The challenger then flips a random bit b. The adversary then wins if it can produce a pre-image x of y such that the first bit of x is b. Clearly, an adversary could always set y to be the hash of an arbitrary x, in which case the first bit of x is b with probability $1/2$. But can the adversary do better?

Classically, the answer is no, assuming the hash is collision resistant. Suppose for a given y that the adversary could win with probability $1/2 + \epsilon$. Then it must win with probability at least ϵ conditioned on $b = 0$, and also with probability at least ϵ conditioned on $b = 1$. By running the adversary on $b = 0$, rewinding until just after the adversary sends y, and running again on $b = 1$, one obtains (with probability at least ϵ^2) pre-images x_0 and x_1 whose first bits are 0,1 respectively. Since $x_0 \neq x_1$ and they are both pre-images of y, we have thus found a collision.

Quantumly, however, the above breaks down. Measuring x_0 on the first execution potentially destroys the quantum state of the adversary, meaning the adversary is no longer guaranteed to produce x_1. Ambainis et al.'s counter-example gives a hash function (relative to an oracle) where the probability to produce x_1 indeed becomes negligible. This creates problems for computationally binding commitments, where Ambainis et al.'s construction yields commitments that are equivocal, despite being binding in the usual sense. Likewise, this equivocation is problematic for many proof systems that demonstrate soundness by extracting two colliding transcripts from an adversary through rewinding.

Unruh's notion of collapsing hashes resolves this problem. Basically, the adversary's first message y results in the output of the hash being measured. Collapsing implies that this is indistinguishable from measuring the input. Measuring the input corresponds exactly to extracting x_0. While such extraction could potentially alter the quantum state, it cannot alter it in any detectable way. In particular this means the second run to recover x_1 must still succeed. This completes the reduction from collision resistance. Note that collision resistance is implied by collapsing as explained by Unruh, and hence collapsing implies the adversary can only win with probability $1/2 + \mathsf{negl}$, as desired.

1.2 Techniques

We call a function $\leq\ell$-to-1 if no image has more than ℓ pre-images. We start with the following observation (Section 3):

Theorem 1 (Informal). *For poly ℓ, any $\leq\ell$-to-1 PQ-CRHF is also collapsing.*

To see why this might be true, consider some $\leq\ell$-to-1 function H. Let

$$|\phi\rangle = \sum_x \alpha_x |x\rangle$$

be a superposition of inputs. Now consider measuring the output of H applied to $|\phi\rangle$ in superposition. If the measurement results in outcome y, then the state

$|\phi\rangle$ collapses to the partially-measured state

$$|\phi_y\rangle \propto \sum_{x:H(x)=y} \alpha_x |x\rangle \ .$$

Since H is $\leq\ell$-to-1, the support of $|\phi_y\rangle$ contains at most ℓ different x.

Non-collapsing means that there is some operation M which distinguishes $|\phi_y\rangle$ from the result of measuring $|\phi_y\rangle$, the latter yielding a distribution over singletons $|x\rangle$ such that $H(x) = y$. Suppose that M actually simply accepted $|\phi_y\rangle$ and rejected all orthogonal states. In this case, if we measure $|\phi_y\rangle$—thus obtaining one pre-image x—and then apply M, there is a non-negligible chance we get back to $|\phi_y\rangle$. This is because $|\phi_y\rangle$ must have a significant overlap with $|x\rangle$, as $|\phi_y\rangle$ is the sum of only ℓ of the $|x\rangle$ vectors. But then if we were to measure again, we will get some x' that is *also* a pre-image. Moreover, $|\phi_y\rangle$ is itself not a singleton, since otherwise measuring it would have no effect and the distinguishing M would be impossible. Therefore there is a non-negligible chance that $x \neq x'$. We thus obtain a collision.

We show that the above actually holds, no matter what $|\phi_y\rangle$ is, and no matter what M does, thus proving Theorem 1.

Generalization. Unfortunately, Theorem 1 appears somewhat limited. One may hope that symmetric hash functions such as SHA2 or SHA3, when restricted to a domain that is only slightly larger than the range, might be $\leq\ell$-to-1 for a polynomial ℓ. After all, if we model them as random oracles, it is straightforward to show this. However, for other hash functions based on post-quantum assumptions, such as LPN [BLVW19, YZW+19] or expanders [CLG09], we cannot reasonably apply the random oracle heuristic due to significant structure. There are two potential problems:

1. The image might be a sparse subset of the co-domain. In this case, even if the hash function only compressed by a single bit, it may be exponentially-many-to-1 and Theorem 1 will not apply. It is not hard to modify Unruh's counterexample [Unr16b] to give such a non-collapsing hash (relative to an oracle). We will give an example of where this is relevant below.
2. Looking ahead, we will see that LPN- and expander-based hash functions will eventually achieve some level of regularity, but this is only guaranteed once the input size is somewhat larger than the output. In such a case, the function is inherently exponentially-many-to-1.

We therefore propose a generalization of Theorem 1 which overcomes these two specific issues above. First, observe that any $\leq\ell$-to-1 hash on its own is not very useful, as it offers only minimal compression. However, by domain extension techniques, we can compile it into a hash function with arbitrary compression.

Imagine using Merkle-Damgård (MD) for domain extension, compiling a "small" hash H into a "big" hash H'. MD is already guaranteed to preserve collapsing [Unr16b]. Imagine at each iteration, we only incorporate a single bit of the input at a time. Since the input to each iteration of H is just an output of

H concatenated with a single bit, the number of possible inputs to H is never more than twice the number of possible outputs. In other words, H is 2-to-1 *on average*, over the set of possible inputs it will be evaluated on. If H were "sufficiently random looking", we would therefore expect that most outputs to H would only have relatively few pre-images, so that H could be $\leq \ell$-to-1 for a polynomial ℓ.

We formalize this intuition: assuming H is "sufficiently regular", we show that we can make H "sufficiently random looking" by pre-pending it with a (almost) ℓ-wise independent permutation for a polynomially-large ℓ. Here, "sufficiently regular" essentially means that the most common output of H is only polynomially-more likely than the average output. This is formalized by a notion we call *semi-regularity* (Definition 4), which says roughly that the most common output is only a polynomial factor more likely than the "average" output. The result is the following:

Theorem 2. *If H is a semi-regular PQ-CRHF, then it can be compiled into a collapsing hash function H'.*

Applications. We show that several candidate post-quantum hash functions satisfy the necessary semi-regularity conditions, thus allowing us to construct novel collapsing hash functions:

- Section 5: Hash functions based on LPN [BLVW19, YZW+19] for a variety of low noise or high-hardness settings, matching the LPN assumptions under which plain post-quantum collision resistance exists.
- Section 6: Hash functions based on walks on exponentially-large expander graphs, as proposed by Charles, Goren, and Lauter [CLG09], abstracting earlier ideas of [TZ94]. A particular instantiation suggested by [CLG09] allows for obtaining a collapsing hash function from the hardness of certain problems on isogenies over elliptic curves. Another candidate was recently proposed by Fuchs et al. [FLLT21] based on Markov Triples.

Remark 1. The output of an expander-based hash is the label of the final node in the walk. In general, the set of labels may be sparse, in which case we would run into Problem 1. An example of such an expander is that of Fuchs et al., where the range is $\mathbb{Z}_p^3$, but the size of the graph is only $O(p^2)$. Likewise, the Charles et al. expander from isogenies has labels in $\mathbb{Z}_p^2$ but the graph size is only $O(p)$. For this reason, in the case of expander hashes, we need the full power of Theorem 2.

Remark 2. We emphasize that we do not prove the constructions of [BLVW19, YZW+19, CLG09, FLLT21] are collapsing. Instead, we only prove semi-regularity, which allows us to compile (through a Merkle-Damgård-like construction) into a collapsing hash. We leave as an interesting open question whether the base constructions could be proven collapsing.

Remark 3. Other instantiations of [CLG09] have been proposed, such as the use of LPS graphs [CLG09], the original proposal of [TZ94], and Morgenstern graphs [PLQ12]. Some weaknesses have been shown in these graphs [PLQ08],

though there are still versions that remain secure. See [PLQ08] for discussion. For any version that is post-quantum collision resistant, our result immediately lifts it to a hash that is collapsing.

Symmetric Key Hash Functions. We do not know how to prove that symmetric hash functions such as SHA2 or SHA3 are semi-regular, and leave this as an interesting open question. However, we observe that random oracles are readily shown to be semi-regular. Thus, either of two things happen:

- The hash function is not semi-regular, therefore violating the random oracle heuristic for a classically defined statistical property. This case could be considered as demonstrating a significant weakness of the hash function.
- The hash function is semi-regular, in which we can compile it into a collapsing hash function based on the assumed (post-quantum) collision resistance of the function, which is a widely studied security property.

Thus we establish semi-regularity as an important design principle in the design of symmetric-key based hash functions.

We also provide additional evidence that SHA2 or SHA3 can be compiled into a collapsing hash. Concretely, SHA2 and SHA3 are widely believed to have *optimal* collision resistance, meaning that any polynomial-time algorithm only has a polynomial advantage over the trivial algorithm of guessing two random inputs and hoping they collide. The assumed optimal collision resistance is the basis for the current parameter settings of these functions. If SHA2 or SHA3 did not have optimal collision resistance, it would show that the parameter settings are too aggressive, and this would be considered a serious weakness.

In Section 7, we show that any optimally (post-quantum) collision resistant hash function that compresses by only a few bits is in fact collapsing, even if it is not semi-regular. Thus under the highly likely optimal collision resistance of SHA2 or SHA3, we obtain a collapsing hash function.

1.3 Collapsing from Group Actions?

A group action is a relaxation of a standard cryptographic group, roughly allowing exponentiation but not multiplication. The advantage of such a restricted structure is that it prevents Shor's algorithm [Sho94], and therefore maintains plausible post-quantum security. This was observed concurrently by Couveignes [Cou06] and Rostovtsev and Stolbunov [RS06], both works also proposing an instantiation of plausible post-quantum group actions using isogenies over elliptic curves.

The restricted structure of group actions preserves plausible post-quantum security, but it also restricts applications. In particular, the usual way of obtaining collision resistance from discrete logarithms, namely

$$(x, y) \mapsto g^x h^y \ ,$$

no longer can be computed without the ability to multiply elements. One could consider another natural construction, namely:

$$(x, b) \mapsto \begin{cases} g^x & \text{if } b = 0 \\ h^x & \text{if } b = 1 \end{cases},$$

where b is a single bit. This is a 2-to-1 function where finding collisions is intractable by the hardness of discrete logarithms on the group action. For group actions based on isogenies, the discrete logarithm problem is exactly the problem of computing isogenies. However, with currently known group actions from isogenies, the bit-length of g^x is roughly *twice* the bit-length of x, meaning the images are sparse and the function is not compressing despite being 2-to-1. Such functions are not useful for hashing. It remains a major open question whether collision resistant compressing hashing can be based on the discrete log problem for group actions of this form, and in particular if such collision resistance can be based on the hardness of computing isogenies.

Call a group action *compact* if g^x has the same bit length as x. For compact group actions, the above hash function would be compressing, and collision resistance would follow from the hardness of computing discrete logarithms. Then applying Theorem 1, we immediately conclude that compact group actions also yield collapsing hash functions.. We leave finding a plausible post-quantum compact group action as an intriguing open question.

Remark 4. The isogeny-based hash of [CLG09] relies on a different problem, namely finding a non-trivial cycle on the isogeny graph. The hardness of finding cycles is a *stronger* assumption that the hardness of computing isogenies.

1.4 Collapsing from Arbitrary Collision Resistance?

While it seems most natural hash functions are semi-regular (at least in some parameter settings), it is not hard to construct contrived hash functions that are not semi-regular. Therefore, our restriction to semi-regular functions potentially limits the applicability of our approach. An interesting conjecture is the following:

Conjecture 1. From *any* PQ-CRHF, one can build a collapsing hash function.

Removing the semi-regularity restriction seems challenging. Consider a construction of H' from H where the output of H' is just the concatenation of t outputs of H on different inputs. More generally, perhaps the output of H' is an injective function applied to t outputs of H. This structure would allow for immediately translating an H' collision into an H collision. It seems difficult to devise an H' that is not of this form while still proving the collision resistance of H' (let alone collapsing) just on the collision resistance of H.

For an H' of this form, if H has n-bit outputs, H' has tn-bit outputs, and therefore H' must have at least $(tn + 1)$-bit inputs in order to be compressing. Suppose H was not semi-regular, and had some outputs that represented an f-fraction of the domain, where f is much larger than the fraction for "average"

outputs, which we will denote g. Then H' will have (information-theoretically) outputs that represent an approximately f^t-fraction of the domain, where the average output would be approximately g^t. Thus H' is not semi-regular, and in fact has even worse regularity if $t > 1$.

Therefore, it seems challenging, if not impossible, to generically remove semi-regularity from a collision resistant hash function. One may hope to prove H' is collapsing despite not being semi-regular. But there would be little hope of using our techniques alone to prove collapsing, since the calls to H could be on inputs mapping to the highly-likely outputs, in which case H is super-poly-to-1.

On the other hand, our situation can be seen as roughly analogous to the case of constructing pseudorandom generators (PRGs) from one-way functions (OWFs). Specifically, Goldreich, Krawczyk, and Luby [GKL88] initially show that PRGs can be constructed from any regular one-way function. This was then improved to PRGs from arbitrary one-way functions by Håstad et al. [HILL99]. Likewise, our hope is that future ideas will allow for proving Conjecture 1.

1.5 Concurrent and Independent Work

In a current and independent work, Cao and Xue [CX22] also study collapsing hash functions. Their core result is identical to Theorem 1, namely that collision resistance when the number of pre-images is polynomially bounded implies collapsing. Somewhat analogous to Theorem 2, they also identify a relaxation they call *almost-regularity*, and show that almost-regular PQ-CRHFs can be used to build collapsing hashes. Almost-regularity is a somewhat stronger requirement than semi-regularity, resulting in fewer applications. [CX22] show that the SIS hash function is almost-regular, thus giving a collapsing hash function from SIS, arriving at the same feasibility result as [Unr16a] though through entirely different means. Our work gives several applications not covered in [CX22], namely collapsing hashes from LPN, expanders, and optimal collision resistance. The former two applications rely on our more general Theorem 2.

2 Preliminaries

Quantum Computation. We give a very brief overview of quantum computation. A pure state is a unit column vector, usually denoted in ket notation as $|\psi\rangle$, in a complex Hilbert space $\mathcal{H}$. The conjugate transpose of $|\psi\rangle$, a row vector, is denoted in bra notation as $\langle\psi|$. We usually think of $\mathcal{H}$ as a product of n 2-dimensional spaces, which are called qubits. For each qubit, we will fix some preferred basis $\{|0\rangle, |1\rangle\}$, which we call the computational basis. An n qubit space is therefore associated with the set of n-bit strings, and we say that $|\psi\rangle$ is a superposition over n-bit strings.

A mixed state is a probability distribution over pure states. If state $|\psi_i\rangle$ occurs with probability p_i, the mixed state is characterized by a density matrix, given by $\sum_i p_i|\psi_i\rangle\langle\psi_i|$. Mixed states are usually denoted as ρ.

A quantum algorithm contains two types of operations: unitary transformations and projective measurements. A unitary is a linear operator U such that $UU^\dagger = \mathbf{I}$, where $U^\dagger$ is the Hermitian transpose. The action of U on $|\psi\rangle$ is given by $U|\psi\rangle$. A projective measurement is specified by a set of projections $\mathcal{P} = (P_1, \ldots, P_t)$ such that $\sum_i P_i = \mathbf{I}$. When applying measurement $\mathcal{P}$ to state $|\psi\rangle$, the result is to output i with probability p_i and the quantum system "collapses" to the state $|\psi_i\rangle$, where:

$$|\psi_i\rangle := \frac{P_i|\psi\rangle}{\sqrt{\langle\psi|P_i|\psi\rangle}} \quad , \quad p_i := \langle\psi|P_i|\psi\rangle \ .$$

When the measurement is applied to a mixed state ρ, the result is to output i with probability p_i and the system collapses to ρ_i, where

$$\rho_i := \frac{1}{p_i} P_i \rho P_i \quad , \quad p_i := \mathsf{Tr}(P_i \rho) \ .$$

For a qubit, measurement in the computational basis is the measurement $(|0\rangle\langle 0|, |1\rangle\langle 1|)$. For a projective measurement $\mathcal{P}$ acting on pure state $|\psi\rangle$ or mixed state ρ, we will write $(i, \rho') \leftarrow \mathcal{P}(|\psi\rangle)$ or $(i, \rho') \leftarrow \mathcal{P}(\rho)$ to denote the output i of applying the measurement $\mathcal{P}$ to ρ, together with the resulting state ρ'. Sometimes we will ignore the actual result of measurement i, focusing just on the resulting state, in which case we write $\rho' \leftarrow \mathcal{P}(|\psi\rangle)$ or $\rho' \leftarrow \mathcal{P}(\rho)$. Other times, we will ignore the resulting state and just focus on the measurement outcome, in which case we write $i \leftarrow \mathcal{P}(|\psi\rangle)$ or $i \leftarrow \mathcal{P}(\rho)$.

Consider a joint system $\mathcal{H} = \mathcal{H}_0 \otimes \mathcal{H}_1$, and applying two measurements $\mathcal{P}_0, \mathcal{P}_1$ to the sub-systems $\mathcal{H}_0, \mathcal{H}_1$. We write the resulting measurement as $\mathcal{P}_0 \otimes \mathcal{P}_1$.

Efficient quantum algorithms are given by a polynomial number of unitaries from some constant-sized universal set and a polynomial number of computational basis measurements. We say such algorithms are *quantum polynomial time* (QPT).

Throughout this work, we will make use of the following fact:

Fact 1. *Any efficient quantum computation over a space $\mathcal{H}$ can be turned into an efficient computation that is also a projective measurement $\mathcal{P}$ over a space $\mathcal{H} \otimes \mathcal{H}'$ for some $\mathcal{H}'$.*

Hash Functions. A hash function will be specified by a family of distributions $\mathcal{H} = (\mathcal{H}_\lambda)_\lambda$ over classically efficiently computable functions $h : \mathcal{X}_\lambda \to \mathcal{Y}_\lambda$ between some domain $\mathcal{X}_\lambda$ and co-domain $\mathcal{Y}_\lambda$. We require non-trivial compression, namely that $|\mathcal{X}_\lambda| \geq 2 \times |\mathcal{Y}_\lambda|$. We will consider two security properties. The first is plain collision resistance but again quantum attackers:

Definition 1 (PQ-CRHF). *$\mathcal{H}$ is a* post-quantum *collision resistant hash function if, for every QPT algorithm $\mathcal{A}$, there exists a negligible function* negl *such that*

$$\Pr\left[\begin{smallmatrix} x_0 \neq x_1, \text{ and} \\ h(x_0) = h(x_1) \end{smallmatrix} : \begin{smallmatrix} h \leftarrow \mathcal{H}_\lambda \\ (x_0, x_1) \leftarrow \mathcal{A}(h) \end{smallmatrix}\right] < \mathsf{negl}(\lambda) \ .$$

The second definition is collapsing, due to Unruh [Unr16b]. Consider a superposition $|\psi\rangle$ over $\mathcal{X}_\lambda$. Consider two measurements:

– $\mathcal{M}_{\mathcal{X}} = (|x\rangle\langle x|)_{x\in\mathcal{X}_\lambda}$, which is just the computational basis measurement of $|\psi\rangle$.
– $\mathcal{M}^h_{\mathcal{Y}} = (\sum_{x:h(x)=y} |x\rangle\langle x|)_{y\in\mathcal{Y}_\lambda}$. This is the measurement corresponding to the following process:
  • First map $|\psi\rangle = \sum_x \alpha_x|x\rangle$ to $|\psi_1\rangle = \sum_x \alpha_x|x\rangle|h(x)\rangle$, a superposition over $\mathcal{X}_\lambda \times \mathcal{Y}_\lambda$.
  • Measure the $\mathcal{Y}_\lambda$ registers to obtain y. The $|\psi_1\rangle$ collapses to a state proportional to $\sum_{x:h(x)=y} \alpha_x|x\rangle|y\rangle$.
  • Discard the $\mathcal{Y}_\lambda$ registers.

The collapsing definition essentially says that, for any superposition of inputs the adversary can produce, if either $\mathcal{M}_{\mathcal{X}}$ or $\mathcal{M}^h_{\mathcal{Y}}$ is applied to the state, it is computationally infeasible to tell which. This holds even if the adversary maintained an arbitrary internal state that could be entangled with the superposition of inputs.

Definition 2 (Collapsing Hash [Unr16b]). *$\mathcal{H}$ is a* collapsing *hash function if, for every QPT algorithm $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$, there exists a negligible function* negl *such that*

$$|\Pr[1 \leftarrow \mathcal{A}_1 \circ (\mathbf{I} \otimes \mathcal{M}_{\mathcal{X}}) \circ \mathcal{A}_0(h)] - \Pr[1 \leftarrow \mathcal{A}_1 \circ (\mathbf{I} \otimes \mathcal{M}^h_{\mathcal{Y}}) \circ \mathcal{A}_0(h)]| < \mathsf{negl}(\lambda) \ ,$$

where both probabilities are over the choice of $h \leftarrow \mathcal{H}_\lambda$. We call the quantity on the left above the advantage *of $\mathcal{A}$. Note that $\mathcal{A}_0$ outputs both a (quantum) internal state and a superposition over $\mathcal{X}_\lambda$. The internal state is passed unaffected to $\mathcal{A}_1$, as is the result of applying $\mathcal{M}_{\mathcal{X}}$ or $\mathcal{M}^h_{\mathcal{Y}}$ to the superposition over $\mathcal{X}_\lambda$.*

Definition 3 (t-wise independence). *A family Π of injections from $\mathcal{X}$ to $\mathcal{Y}$ ($|\mathcal{Y}| \geq |\mathcal{X}|$) is a t-wise δ-dependent injection if, for any distinct $x_1, \ldots, x_t \in \mathcal{X}$, the distribution $(\pi(x_1), \ldots, \pi(x_t))$ for $\pi \leftarrow \Pi$ is δ-close to t uniformly random* distinct *elements of $\mathcal{Y}$.*

Distributions and Rényi Entropy. For a distribution D over a finite set I, and $\alpha > 1$, define the *Rényi Entropy* as

$$H_\alpha(D) := -\frac{1}{\alpha - 1} \log \left(\sum_{i\in I} \Pr[i \leftarrow D]^\alpha \right)$$
$$H_\infty(D) := -\log \max_{i\in I} \Pr[i \leftarrow D]$$

The choice of base in the logarithm is irrelevant for our purposes, as long as the same base is used for all α. For our purposes, it will be convenient to map Rényi entropy to the norm of the probability vector. Write

$$\|D\|_\alpha := \left(\sum_{i\in I} \Pr[i \leftarrow D]^\alpha \right)^{1/\alpha} = 2^{-\left(1-\frac{1}{\alpha}\right) H_\alpha(D)}$$
$$\|D\|_\infty := \max_{i\in I} \Pr[i \leftarrow D] = 2^{-H_\infty(D)}$$

For $\beta > \alpha \geq 1$, we have the following inequalities, where the left and right inequalities are identical just phrased in terms of entropies vs vector norms:

$$H_\alpha(D) \geq H_\beta(D) \qquad (\|D\|_\alpha)^{\frac{\alpha}{\alpha-1}} \leq (\|D\|_\beta)^{\frac{\beta}{\beta-1}} \tag{1}$$

$$\left(1 - \frac{1}{\alpha}\right) H_\alpha(D) \leq \left(1 - \frac{1}{\beta}\right) H_\beta(D) \qquad \|D\|_\alpha \geq \|D\|_\beta \tag{2}$$

$$H_\alpha(D) \leq \log |I| \qquad \|D\|_\alpha \geq |I|^{-1} \tag{3}$$

Let $\Delta_\alpha(D) := H_\alpha(D) - H_\infty(D)$ to be the *Entropy Gap* of D. When α is not specified, we will mean $\alpha = 2$.

For a finite set $\mathcal{X}$, we abuse notation and use $\mathcal{X}$ to denote the uniform distribution over $\mathcal{X}$. For a function $h : \mathcal{X} \to \mathcal{Y}$ and a distribution D on $\mathcal{X}$, we let $h(D)$ be the distribution obtained by sampling $x \leftarrow D$ and then outputting $h(x)$. We also define $H_\alpha(h) := H_\alpha(h(\mathcal{X}))$, $\|h\|_\alpha := \|h(\mathcal{X})\|_\alpha$, and $\Delta_\alpha(h) := \Delta_\alpha(h(\mathcal{X}))$.

3 From Non-Collapsing to Equivocation

Here, we prove that a failure to be collapsing leads to equivocation. We consider the following setup:

- A secret set S of size ℓ, which is a subset of some set $\mathcal{U}$.
- Another set $\mathcal{V}$.
- A state ρ that is a superposition over pairs $(v, s) \in \mathcal{V} \times S$.
- A binary-outcome projective measurement $\mathcal{P} = (\mathbf{P}, \mathbf{I} - \mathbf{P})$.

Our goal is to, starting in the state ρ, obtain two distinct values $i, j \in S$. The only operations we can perform are the measurement $\mathcal{P}$ and the measurement in the computational basis for $\mathcal{U}$. Without any further promises, this goal is impossible. By applying $\mathcal{U}$ to ρ, one obtains a single element of S. If $\mathcal{P}$, say, commutes with $\mathcal{U}$, then no sequence of operations will ever change the state, and we will never obtain a second element.

Therefore, we are given the promise that $\mathcal{P}$ is sufficiently non-commuting with $\mathcal{U}$. Concretely, we are promised that:

$$|\Pr[1 \leftarrow \mathcal{P}(\rho)] - \Pr[1 \leftarrow (\mathcal{P} \circ (\mathbf{I} \otimes \mathcal{U}))(\rho)]| \geq \epsilon$$

for some non-negligible quantity ϵ. In other words, $\mathcal{P}$ distinguishes between ρ and the result of measuring ρ in the computational basis for $\mathcal{U}$.

The Algorithm. Since we are now only allowed to use $\mathcal{U}$ and $\mathcal{P}$, there is nothing that can be done except alternate them. Concretely, we apply $\mathcal{U}$, $\mathcal{P}$, and then $\mathcal{U}$ again. We will show that, with non-negligible probability, the two applications of $\mathcal{U}$ output distinct elements of S.

Lemma 1. *For $\ell, S, \rho, \mathcal{P}, \mathcal{U}, \mathcal{V}$ as defined above,*

$$\Pr\left[\begin{smallmatrix} i,j \in S \\ i \neq j \end{smallmatrix} : \begin{smallmatrix} (i,\rho') \leftarrow (\mathbf{I} \otimes \mathcal{U})(\rho) \\ \rho'' \leftarrow \mathcal{P}(\rho') \\ j \leftarrow (\mathbf{I} \otimes \mathcal{U})(\rho'') \end{smallmatrix}\right] \geq \frac{2}{\ell - 1} \left| \begin{smallmatrix} \Pr[1 \leftarrow \mathcal{P}(\rho)] \\ - \Pr[1 \leftarrow (\mathcal{P} \circ (\mathbf{I} \otimes \mathcal{U}))(\rho)] \end{smallmatrix} \right|^2 .$$

Before proving Lemma 1, we observe that it is tight. Let q be the quantity on the left, and r the quantity inside $|\cdot|$ on the right. Consider the case where $\mathcal{V}$ is empty, ρ is the pure state $|\psi\rangle := \ell^{-1/2}\sum_{i\in S}|i\rangle$, and $\mathbf{P}$ is the projection onto $|\psi\rangle$. In this case, Applying $\mathcal{P}$ to $|\psi\rangle$ outputs 0 with certainty. Meanwhile, measuring $|\psi\rangle$ gives a random $|i\rangle$, and applying $\mathcal{P}$ to any $|i\rangle$ will give 0 with probability $1/\ell$. Therefore, $r = 1 - 1/\ell$, and the right-hand side becomes $2(\ell-1)/\ell^2$.

On the other hand, for computing q, there are two cases: (1) if applying $\mathcal{P}$ to $|i\rangle$ outputs 0, or (2) it outputs 1. If it outputs 0 (which occurs with probability $1/\ell$), then the state is back to $|\psi\rangle$, and measuring again will give an $j \neq i$ with probability $1 - 1/\ell$. If it outputs 0 (which occurs with probability $1 - 1/\ell$), then the state becomes $|i\rangle - \ell^{-1/2}|\psi\rangle$. In this case, a simple calculation shows that measurement will give $j \neq i$ with probability $1/\ell$. Taken together, the overall probability q of obtaining a $j \neq i$ is exactly $2(\ell-1)/\ell^2$, exactly matching the right-hand side.

We now give the proof of Lemma 1.

Proof. We focus on the case of pure states, the mixed state setting then following from convexity. Therefore we assume $\rho = |\psi\rangle\langle\psi|$ for some pure state $|\psi\rangle = \sum_{v,i}\alpha_{v,i}|v,i\rangle$.

We first analyze q. The probability of obtaining i in the first measurement is $p_i = \mathsf{Tr}\left[(\mathbf{I}\otimes|i\rangle\langle i|)\rho\right]$, in which case ρ' becomes $\rho_i := \frac{1}{p_i}(\mathbf{I}\otimes|i\rangle\langle i|)\rho(\mathbf{I}\otimes|i\rangle\langle i|)$.

Now we apply $\mathcal{P}$, and disregard the output of the measurement. The resulting mixed state is $\rho_i' := \mathbf{P}\rho_i\mathbf{P} + (\mathbf{I}-\mathbf{P})\rho_i(\mathbf{I}-\mathbf{P})$. Now we apply $(\mathbf{I}\otimes\mathcal{U})$ again. The probability of obtaining j is $\mathsf{Tr}\left[(\mathbf{I}\otimes|j\rangle\langle j|)\rho_i'\right]$. Summing over all $i \in S$ and $j \in S\setminus\{i\}$, we have that the probability of obtaining distinct $i, j \in S$ is q where

$$
\begin{aligned}
q &= \mathsf{Tr}\left[\sum_{i,j\in S, i\neq j} \begin{array}{c}(\mathbf{I}\otimes|j\rangle\langle j|)\mathbf{P}(\mathbf{I}\otimes|i\rangle\langle i|)\rho(\mathbf{I}\otimes|i\rangle\langle i|)\mathbf{P} \\ +(\mathbf{I}\otimes|j\rangle\langle j|)(\mathbf{I}-\mathbf{P})(\mathbf{I}\otimes|i\rangle\langle i|)\rho(\mathbf{I}\otimes|i\rangle\langle i|)(\mathbf{I}-\mathbf{P})\end{array}\right] \\
&= 2\mathsf{Tr}\left[\sum_{i,j\in S, i\neq j}(\mathbf{I}\otimes|j\rangle\langle j|)\mathbf{P}(\mathbf{I}\otimes|i\rangle\langle i|)\rho(\mathbf{I}\otimes|i\rangle\langle i|)\mathbf{P}\right] \\
&= 2\mathsf{Tr}\left[\sum_{\substack{i,j\in S, i\neq j \\ v,v'\in\mathcal{V}}}\alpha_{v,i}\alpha_{v',i}^{\dagger}(\mathbf{I}\otimes|j\rangle\langle j|)\mathbf{P}(|v\rangle\langle v'|\otimes|i\rangle\langle i|)\mathbf{P}\right] \\
&= 2\left[\sum_{\substack{i,j\in S, i\neq j \\ v,v'\in\mathcal{V}}}\alpha_{v,i}\alpha_{v',i}^{\dagger}(\langle v'|\langle i|)\mathbf{P}(\mathbf{I}\otimes|j\rangle\langle j|)\mathbf{P}(|v\rangle|i\rangle)\right] \\
&= 2\left[\sum_{\substack{i,j\in S, i\neq j \\ v,v',v''\in\mathcal{V}}}\alpha_{v,i}\alpha_{v',i}^{\dagger}\langle v',i|\mathbf{P}|v'',j\rangle\ \langle v'',j|\mathbf{P}|v,i\rangle\right] .
\end{aligned}
$$

Then if we define $\mathbf{w}$ as the vector indexed by tuples $(i,j,v''), i \neq j$ such that $\mathbf{w}_{(i,j,v'')} := \sum_v \alpha_{v,i}\langle v'', j|\mathbf{P}|v, i\rangle$, we have that $q = 2|\mathbf{w}|^2$.

Next we analyze the right hand side, r, of Lemma 1. We have

$$\begin{aligned} r &= \mathsf{Tr}\left[\mathbf{P}\rho\right] - \mathsf{Tr}\left[\mathbf{P}\sum_{i\in S}(\mathbf{I}\otimes|i\rangle\langle i|)\rho(\mathbf{I}\otimes|i\rangle\langle i|)\right] \\ &= \left[\sum_{\substack{i,j\in S\\ v,v'\in\mathcal{V}}} \alpha_{v,i}\alpha^\dagger_{v',j}\langle v'|\langle i|\mathbf{P}|v\rangle|j\rangle - \sum_{\substack{i\in S\\ v,v'\in\mathcal{V}}} \alpha_{v,i}\alpha^\dagger_{v',i}\langle v'|\langle i|\mathbf{P}|v\rangle|i\rangle\right] \\ &= \left[\sum_{\substack{i,j\in S, i\neq j\\ v,v'\in\mathcal{V}}} \alpha_{v,i}\alpha^\dagger_{v',j}\langle v'|\langle i|\mathbf{P}|v\rangle|j\rangle\right] \quad . \end{aligned}$$

Then if we define $\mathbf{x}$ as the vector $\mathbf{x}_{(i,j,v'')} := \alpha_{v'',j}$, we have that $r = \mathbf{x}\cdot\mathbf{w}$. Note that

$$|\mathbf{x}|^2 = \sum_{\substack{i,j\in S, i\neq j\\ v''\in\mathcal{V}}} |\alpha_{v'',j}|^2 = \sum_{j\in S, v''\in\mathcal{V}} (\ell-1)|\alpha_{v'',j}|^2 = \ell - 1 \quad .$$

Therefore, by the Cauchy-Schwartz inequality, we have that $|\mathbf{w}|^2|\mathbf{x}|^2 \geq |\mathbf{w}\cdot\mathbf{x}|^2$. The lemma follows. □

3.1 Application: Hashing with small compression.

We now use Lemma 1 to show that any hash function which is $\leq\ell$-to-1 for a polynomial ℓ is collapsing.

Theorem 1. *Let $\mathcal{H}$ be a post-quantum collision-resistant hash function with domain $\mathcal{X}$, and ℓ a polynomial. Suppose that, with overwhelming probability over the choice of $h \leftarrow \mathcal{H}$, that h is $\leq\ell$-to-1. Then $\mathcal{H}$ is collapsing.*

Proof. Assume toward contradiction that $\mathcal{H}$ is not collapsing. Let $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ be the adversary for the collapsing game, with non-negligible advantage ϵ. We will think of $\mathcal{A}_1$ as being a projective measurement on the joint system $\mathcal{V}\times\mathcal{X}_\lambda$, where $\mathcal{V}$ is the adversary's internal state.

Observe that $\mathcal{M}_\mathcal{X}$ is equivalent to the composition of $\mathcal{M}_\mathcal{Y}$ followed by $\mathcal{M}_\mathcal{X}$, since the domain element uniquely determines the range element. Therefore, we can think of both sides of the collapsing experiment as applying $\mathcal{M}_\mathcal{Y}$, and then the only difference is whether an additional $\mathcal{M}_\mathcal{X}$ is applied. We will therefore always think of the output of $\mathcal{A}_0$ as having $\mathcal{M}_\mathcal{Y}$ applied.

For a fixed h and result y from $\mathcal{M}_\mathcal{Y}$, suppose $\mathcal{A}_1$ has a distinguishing advantage ϵ_h. Then we can apply Lemma 1 to extract two pre-images of y (and hence a collision) with probability at least $2\epsilon_h^2/(\ell-1)$. By averaging over all h and y and invoking convexity, we see that the overall probability of finding a collision is at least $2\epsilon^2/(\ell-1)$, which is non-negligible. □

By combining with the fact that standard domain extension works for collapsing hash functions, we have the following corollary:

Corollary 1. *Assuming the existence of $\leq\ell$-to-1 PQ-CRHFs for a polynomial ℓ, there exist collapsing hash functions for arbitrary domains.*

4 The Main Theorem

We now generalize the $\leq\ell$-to-1 case to a somewhat more general class of hash functions. The main challenge, of course, is that general hash functions may not be $\leq\ell$-to-1 for any polynomial ℓ. This can be a problem *even if* the domain is only slightly larger than the co-domain. Here, we show how to somewhat relax the conditions on the hash function.

Definition 4. *Let $\mathcal{H} = (\mathcal{H}_\lambda)_\lambda$ be a family of hash functions with domain $\mathcal{X}_\lambda$ and co-domain $\mathcal{Y}_\lambda$. We say that $\mathcal{H}$ is* semi-regular *if there exists a polynomial r and negligible* negl *such that*

$$\Pr_{h\leftarrow\mathcal{H}_\lambda}[\Delta_2(h) > \log r(\lambda)] < \mathsf{negl}(\lambda) \ .$$

Equivalently, $\|h\|_\infty \leq r(\lambda) \times \|h\|_2^2$, except with negligible probability.

For a function h, we will call $\|h\|_\infty/\|h\|_2^2$ the *regularity* of h. A semi-regular hash function is therefore one where the regularity is a polynomial except with negligible probability.

Main Theorem. We now give our main theorem.

Theorem 2. *If there exists a semi-regular PQ-CRHF, then there exists a collapsing hash function.*

The remainder of this section is devoted to proving Theorem 2. We start by considering the following hash function:

Construction 1. *Let $\mathcal{H}$ be a family of post-quantum collision resistant hash functions with domain $\mathcal{X}_\lambda$ and co-domain $\mathcal{Y}_\lambda$. For parameters $\ell \in \mathbb{Z}, \delta \in [0,1]$, let $\mathcal{F}$ be a ℓ-wise δ-dependent injection with domain $\mathcal{Y}_\lambda \times \{0,1\}$ and co-domain $\mathcal{X}_\lambda$. Then for any polynomial $m = m(\lambda)$, we construct the following function family $\mathcal{H}'$ with domain $\{0,1\}^m$ and co-domain $\mathcal{Y}_\lambda$, where $h' \leftarrow \mathcal{H}'$ is sampled as follows: sample $h \leftarrow \mathcal{H}$ and for $i = 1, \ldots, m \times t$, sample $f_i \leftarrow \mathcal{F}$, where t is a parameter to be specified later. Also fix an arbitrary $y_0 \in \mathcal{Y}_\lambda$. Then output $h' : \{0,1\}^m \to \mathcal{Y}_\lambda$ defined as:*

- *For $i = 1, \ldots, u = (m-1) \times t + 1$:*
  - *Let $z_i = y_{i-1}||x_j$ if $i = t(j-1)+1$, otherwise let $z_i = y_{i-1}||0$.*
  - *Let $y_i = h(f_i(z_i))$*
- *Output y_u*

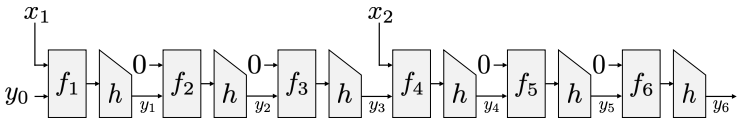

Fig. 1: The first few iterations of Construction 1 for $t = 3$.

The operation of h' is also given in Figure 1.

Remark 5. Note that Construction 1 is only defined for a bounded domain, since it needs independent f_i for each application of h. However, we can set m to be large enough so that $2^m \gg \mathcal{Y}_\lambda$, obtaining a compressing collapsing function. Then we can plug the result into a plain Merkle-Damgård or other domain extender, which are known to preserve collapsing [Unr16a]. The result is an arbitrary-domain hash function that is collapsing.

Remark 6. Observe that some iterations of Construction 1 incorporate bits of the input into the z_i, while others just incorporate 0's. This is mostly an artifact of our proof of collapsing, and it is unclear if it is strictly needed. Looking ahead, in each iteration that incorporates an input bit, the number of possible z_i values potentially doubles, while in other iterations, we show that the number of possible z_i values decreases with noticeable probability. By inserting sufficiently many 0 iterations, we can make sure the number of possible z_i values never gets too large, which we can then use to apply Lemma 1.

For the remainder of the proof, we omit λ subscripts and write $\mathcal{X} = \mathcal{X}_\lambda$ and $\mathcal{Y} = \mathcal{Y}_\lambda$ to keep notation simple. Let $\mathcal{Y}_i$ be the set of possible values for z_i as x ranges over all possible inputs, and $N_i = |\mathcal{Y}_i|$. Let M_i be the number of possible values for y_i. Observe that $N_i = 2M_i$ for $i = t(j-1)+1$ and $N_i = M_i$ otherwise. Define the following quantities:

$$r = \|h\|_\infty / \|h\|_2^2 \tag{4}$$

$$\ell = \max(2re, 3\log|\mathcal{Y}|) \tag{5}$$

$$\delta = |\mathcal{Y}|^{-2} \binom{|\mathcal{Y}|}{\ell}^{-1} \tag{6}$$

$$t = 200\ell \tag{7}$$

Lemma 2. *Except with negligible probability over the choice of h, f_i, the following hold:*

- *$N_i \le \|h\|_2^{-2}$ for all i*
- *For all i, the function $h_i(y) = h(f_i(y))$, when restricted to $\mathcal{Y}_{i-1}$, is $< \ell$-to-1.*

Before proving Lemma 2, we first demonstrate that it allows for proving Construction 1 is collapsing. Note that only the second bullet is needed to prove collapsing; the first bullet is facilitates our proof of Lemma 2 by induction.

Construction 1 is just Merkle-Damgård, composed of u functions $h_i(y) = h(f_i(y))$, where each h_i has domain $\mathcal{Y}_{i-1}$ and the input to the hash has a number of zeros inserted between the various input bits. Each of the h_i are collision resistant since the f_i are injective. By Lemma 2, each of the h_i are also $< \ell$-to-1 when restricting to the set of possible inputs. Hence by Theorem 1, each of the h_i are collapsing on their restricted domains. Unruh [Unr16a] shows that Merkle-Damgård is collapsing if the component h_i are collapsing, hence Construction 1 is collapsing. The exact same proof works here, the only difference is that the h_i are only collapsing on the outputs of h_{i-1}, but are potentially not collapsing on the entire domain $\mathcal{Y}_\lambda \times \{0,1\}$. Nevertheless the same proof works here: imagine y_u is measured. Now measure z_u, then z_{u-1}, then z_{u-2}, etc, until we measure z_1. The application of each measurement is undetectable by the collapsingness of the h_i on their restricted domains. By the time we have measured all of the z_i's, we have measured the entire input. Hence measuring y_u (the output of h') is indistinguishable from measuring the input x.

For completeness, we work out the proof here. We need to show that measuring the final output y_u vs measuring the input x is computationally indistinguishable. We will do this through a hybrid argument. Let $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ be a a collapsing adversary for $\mathcal{H}'_0$, where the probability of distinguishing the measurement $\mathcal{M}^{h'}_{\mathcal{Y}}$ from $\mathcal{M}_{\mathcal{X}}$ is a non-negligible ϵ.

Consider evaluating h' on a quantum superposition, writing the output y_u to a new register Y_u. During iteration j, a number of intermediate values will be stored in a register, including z_j which will be stored in a register Z_j. After the final output y_u of h' is produced and written to a register Y_u, all the intermediate registers including the Z_j will be uncomputed.

In Hybrid i, register Y_u is measured to give y_u, and also registers Z_j for $j = i, \ldots, u$ are all measured before uncomputation, giving z_j. Let p_i be the probability $\mathcal{A}$ outputs 1 in Hybrid i.

Hybrid $u+1$ means none of the Z_j registers are measured, whereas in Hybrid 1, all of the Z_j are measured, which is equivalent to measuring the input registers. Thus $|p_1 - p_{u+1}| = \epsilon$, by our assumption that $\mathcal{A}$ is a collapsing adversary. For each i, we obtain a collapsing adversary $\mathcal{B}^{(i)} = (\mathcal{B}^{(i)}_0, \mathcal{B}^{(i)}_1)$ for h_i with advantage $\epsilon_i = |p_i - p_{i+1}|$. $\mathcal{B}^{(i)}_0(h_i)$ works as follows:

- It first chooses f_j for $j \neq i$, and constructs h' as above. Then it simulates $\mathcal{A}_0(h')$.
- $\mathcal{A}_0$ produces $\rho_{\mathsf{state},X}$, where state is a register containing the adversary's state that gets forwarded to the next stage, and X is a register containing a superposition of inputs to h'.
  $\mathcal{B}^{(i)}_0$ evaluates h' on register X, and measures the registers $Z_{i+1}, \ldots, Z_u$. During the uncomputation step, it uncomputes Y_u and all the registers containing all the intermediate values, *except* for the register Z_i.
- $\mathcal{B}^{(i)}_0$ then outputs the joint system $\rho_{\mathsf{state}',Z_i}$, where $\mathsf{state}' = (\mathsf{state}, X)$.

$\mathcal{B}^{(i)}_1$, upon receiving $\rho_{\mathsf{state}',Z_i}$, uncomputes the Z_i registers, obtaining the system $\rho_{\mathsf{state}'} = \rho_{\mathsf{state},X}$, which it feeds into $\mathcal{A}_1$. It outputs whatever $\mathcal{A}_1$ outputs.

Since $\mathcal{B}_0^{(i)}$ measures register Z_{i+1} to obtain z_{i+1} which includes $y_i = h_i(z_i)$, if the challenger for $\mathcal{B}^{(i)}$ measures the output of h_i, the measurement is redundant and has no effect on the state. Therefore, $\mathcal{B}^{(i)}$ perfectly simulates Hybrid $i+1$. On the other hand, if the challenger measures the input, this is exactly the same as measuring Z_i to obtain z_i. Hence $\mathcal{B}^{(i)}$ perfectly simulates Hybrid i in this case. Therefore, $\mathcal{B}^{(i)}$ has advantage exactly $\epsilon_i = |p_i - p_{i+1}|$.

We then turn each $\mathcal{B}^{(i)}$ into a collision-finder for h, which we call $\mathcal{C}^{(i)}$, following Theorem 1. Conditioned on Lemma 2 holding, the functions h_i are $<\ell$-to-1, meaning $\mathcal{C}^{(i)}$ finds a collision with probability at least $2\epsilon_i^2/(\ell-1)$. Notice that $\sum_i \epsilon_i \geq \epsilon$. Therefore, we can obtain an overall collision-finder $\mathcal{C}$, which runs $\mathcal{C}^{(i)}$ for a random choice of i. By Cauchy-Schwartz, the probability $\mathcal{C}$ obtains a collision is at least

$$\frac{2}{u(\ell-1)} \sum_i \epsilon_i^2 \geq \frac{2\epsilon^2}{u^2(\ell-1)} \ ,$$

which is non-negligible. This contradicts the assumed collision resistance of h.

We now turn to proving Lemma 2.

Proof. We prove by induction on i. Clearly $N_0 = 2$ and h_1 is at most 2-to-1. We now fix h and $f_1, \ldots, f_{i-1}$, which determines $\mathcal{Y}_{i-1}$ and N_{i-1}. We inductively assume $N_{i-1} \leq \|h\|_2^{-2}$. We first prove, with overwhelming probability over the choice of f_i, that h_i is $\leq \ell$-to-1 when restricted to $\mathcal{Y}_{i-1}$.

Toward that end, for any $y \in \mathcal{Y}$, let p_y be the probability a random input to h maps to y. For any set of ℓ inputs $x_1, \ldots, x_\ell$, the probability they all map to the same output of h is:

$$\begin{aligned}
\Pr[h_i(x_1) = \cdots = h_i(x_\ell)] &\leq \Pr_{\substack{w_j \leftarrow \mathcal{X} \\ w_{j_1} \neq w_{j_2} \forall j_1 \neq j_2}} [h(w_1) = \cdots = h(w_\ell)] + \delta \\
&\leq \Pr_{w_j \leftarrow \mathcal{X}} [h(w_1) = \cdots = h(w_\ell)] + \delta \\
&= \sum_{y \in \mathcal{Y}} p_y^\ell + \delta = \|h\|_\ell^\ell + \delta
\end{aligned}$$

Let V be the event that h_i is *not* $<\ell$-to-1. Union-bounding over all sets of ℓ inputs in $\mathcal{Y}_{i-1}$, we have that

$$\begin{aligned}
\Pr[V] &\leq \binom{N_{i-1}}{\ell} (\|h\|_\ell^\ell + \delta) \\
&\leq \frac{N_{i-1}^\ell \|h\|_\ell^\ell}{\ell!} + \delta \binom{N_{i-1}}{\ell} \\
&\leq \frac{N_{i-1}^\ell \|h\|_\infty^{\ell-1}}{\ell!} + |\mathcal{Y}|^{-2} && \text{Equations (1) and (6)} \\
&\leq \frac{(N_{i-1}\|h\|_\infty)^\ell \|h\|_\infty^{-1}}{\ell!} + |\mathcal{Y}|^{-2} \\
&\leq \frac{(N_{i-1} r \|h\|_2^2)^\ell |\mathcal{Y}|}{\ell!} + |\mathcal{Y}|^{-2} && \text{Equation (4)}
\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{r^\ell |\mathcal{Y}|}{\ell!} + |\mathcal{Y}|^{-2} && \text{Inductive assumption} \\
&\leq \left(\frac{re}{\ell}\right)^\ell |\mathcal{Y}| + |\mathcal{Y}|^{-2} && \text{Stirling's Approximation} \\
&\leq 2^{-\ell}|\mathcal{Y}| + |\mathcal{Y}|^{-2} = 2 \times |\mathcal{Y}|^{-2} && \text{Equation (5)}
\end{aligned}
$$

$|\mathcal{Y}|$ must be superpolynomial by the assumed collision resistance of h, and so the above quantity is negligible. Now it remains to prove the desired size bounds. First recall that $N_{t(j-1)+1} \leq 2N_{t(j-1)}$ and $N_i \leq N_{i-1}$ for all i not of the form $t(j-1)+1$. The following suffices to prove the size bound in Lemma 2:

Claim. $N_{t(j-1)} \leq \|h\|_2^{-2}/2$ for all j.

This claim implies that $N_{t(j-1)+1} \leq \|h\|_2^{-2}$, and therefore all $N_{t(j-1)+k} \leq \|h\|_2^{-2}$ for all $k = 2, \ldots, t$, thus proving Lemma 2. We now prove the claim by induction. Clearly for $j = 1$ we have that $N_{t(j-1)} = N_0 = 1$, which is $\leq \|h\|_2^{-2}/2$ since $\|h\|_2^2$, the collision probability of two random inputs to h, must be negligible. This establishes the base case.

We now inductively assume that $N_{t(j-1)+1} \leq \|h\|_2^{-2}$. Our goal is to prove that $N_{t(j-1)+t} \leq \|h\|_2^{-2}/2$. Note that if *any* i in the interval $t(j-1)+2, \ldots, tj$ satisfy $N_i \leq \|h\|_2^{-2}/2$, then we are done since all subsequent i in the interval have $N_i \leq N_{i-1}$. From now on, we will therefore assume towards contradiction that $N_i > \|h\|_2^{-2}/2$ for all i in the interval.

Let C_i be the number of distinct pairs of colliding inputs to h_i. We observe the following:

Claim. If h_i is $<\ell$-to-1, then $M_i < N_{i-1} - \frac{2}{\ell}C_i$.

The claim is proved as follows: by linearity, it suffices to consider the case where h_i has a single output, meaning $M_i = 1$ and $N_i < \ell$. In this case, we have that

$$
\begin{aligned}
N_{i-1} - \frac{2}{\ell}C_i = N_{i-1} - \frac{2}{\ell}\binom{N_{i-1}}{2} &= N_{i-1} - \frac{N_{i-1}}{\ell}(N_{i-1}-1) \\
&> N_{i-1} - (N_{i-1}-1) = 1 = M_i \ .
\end{aligned}
$$

Therefore, to bound $N_i = M_i$ for $i = t(j-1)+2, \ldots, tj$, we need to bound C_i. To do so, let P_2 be the probability that two random distinct inputs to h map to the same image. Then

$$
P_2 = \sum_y p_y \left(\frac{p_y|\mathcal{X}|-1}{|\mathcal{X}|-1}\right) = \frac{|\mathcal{X}|\|h\|_2^2 - 1}{|\mathcal{X}|-1} \geq \|h\|_2^2 - |\mathcal{X}|^{-1} \ .
$$

For a set $L \subseteq \mathcal{Y}_{i-1}$, let E_L be the indicator function for the event that all L map to the same value under h_i. Then $C_i = \sum_{L \subseteq \mathcal{Y}_{i-1}:|L|=2} E_L$. We now calculate the mean of C_i:

$$
\mathbb{E}[C_i] = \sum_{L \subseteq \mathcal{Y}_{i-1}:|L|=2} \mathbb{E}[E_L] \geq \sum_{L \subseteq \mathcal{Y}_{i-1}:|L|=2} (P_2 - \delta) \geq \binom{N_{i-1}}{2} P_2 - 1
$$

$$\geq \binom{N_{i-1}}{2}(\|h\|_2^2 - |\mathcal{X}|^{-1}) - 1 = \frac{N_{i-1}^2 - N_{i-1}}{2}(\|h\|_2^2 - |\mathcal{X}|^{-1}) - 1$$
$$\geq \frac{N_{i-1}^2}{2}(\|h\|_2^2 - |\mathcal{X}|^{-1}) - 2 \ .$$

Recall that $\|h\|_2^2 \geq |\mathcal{Y}|^{-1} \geq 2|\mathcal{X}|^{-1}$ and that $N_{i-1}\|h\|_2^2 \in (1/2, 1]$ by assumption. Therefore, $\mathbb{E}[C_i] \geq N_{i-1}/8 - 2$.

From above we know that $\Pr[V] \leq 2|\mathcal{Y}|^{-2}$. Now we have, for $i = t(j-1) + 2, \ldots, tj$ and assuming each such $N_{i-1} > \|h\|_2^{-2}$,

$$\begin{aligned}
\mathbb{E}[N_i] &= \mathbb{E}[N_i|\neg V](1 - \Pr[V]) + \mathbb{E}[N_i|V]\Pr[V] \\
&\leq \mathbb{E}[N_{i-1} - (2/\ell)C_i|\neg V](1 - \Pr[V]) + N_{i-1}\Pr[V] \\
&\leq (N_{i-1} - (2/\ell)\,\mathbb{E}[C_i|\neg V])(1 - \Pr[V]) + N_{i-1}\Pr[V] \\
&\leq N_{i-1} - (2/\ell)(\mathbb{E}[C_i] - \mathbb{E}[C_i|V]\Pr[V]) + N_{i-1}\Pr[V] \\
&\leq N_{i-1} - (2/\ell)(N_{i-1}/8 - 2) + (2/\ell\,\mathbb{E}[C_i|V] + N_{i-1})\Pr[V] \\
&\leq N_{i-1} - (2/\ell)(N_{i-1}/8 - 2) + N_{i-1}^2\Pr[V] \\
&\leq N_{i-1} - (2/\ell)(N_{i-1}/8 - 2) + 2 \\
&\leq N_{i-1} - N_{i-1}/5\ell
\end{aligned}$$

Since N_i is between 1 and N_{i-1}, we must have that

$$\Pr[N_i < N_{i-1}(1 - 1/10\ell)] \geq 1/10\ell \ .$$

Call an i "good" $N_i < N_{i-1}(1 - 1/10\ell)$. Let T be the number of good i. Suppose there are $\geq T$ good i in the interval $t(j-1) + 2, \ldots, tj$. Then $N_{tj} < (1 - 1/10\ell)^{10\ell} N_{t(j-1)+1} \leq (e^{-1} - o(1))\|h\|_2^{-2} \leq \|h\|_2^{-2}/2$. Since we assumed this was not the case, it must be that $T < 10\ell$. But $\mathbb{E}[T] \geq t/10\ell = 20\ell$, so by Hoeffding's inequality,

$$\Pr[T < 10\ell] \leq \Pr[T - \mathbb{E}[T] < -10\ell] < e^{-2(10\ell)^2/t} = e^{-\ell} \ .$$

Thus, except with negligible probability, N_{tj} must in fact be $\leq \|h\|_2^{-2}/2$. This completes the proof of Lemma 2 and hence Theorem 2. □

5 Collapsing Hashes from LPN

In this section, we construct collapsing hash functions from the hardness of learning parities with noise (LPN) in certain extreme parameter regimes.

5.1 LPN-Based Hashing

For positive integers $n, m > n$ and error rate $\epsilon \in [0, 0.5]$, define $\mathsf{LPN}_\epsilon^{n\times m}$ to be the following distribution: choose a random $s \leftarrow \mathbb{Z}_2^n$ and random $A \leftarrow \mathbb{Z}_2^{n\times m}$. Choose a random $e \in B_\epsilon^m$, B_ϵ is the Bernoulli distribution: output 1 with probability ϵ and 0 otherwise. The output of $\mathsf{LPN}_\epsilon^{n\times m}$ is then $(A, s^T \cdot A + e^T \bmod 2)$. The LPN assumption states that it is computationally infeasible to distinguish $\mathsf{LPN}_\epsilon^{n\times m}$ from the uniform distribution $\mathbb{Z}_2^{(n+1)\times m}$. Specifically:

Assumption 1. *For parameters $\epsilon = \epsilon(n), m = m(n), T = T(n)$, The (ϵ, m, T)-LPN assumption is that, for any adversary $\mathcal{A}$ running in time at most T, there exists a negligible $\mathsf{negl}(n)$ such that*
$|\Pr[1 \leftarrow \mathcal{A}(\mathsf{LPN}_\epsilon^{n\times m})] - \Pr[1 \leftarrow \mathcal{A}(\mathbb{Z}_2^{(n+1)\times m})]| < \mathsf{negl}(n)$.

Brakerski et al.[BLVW19] and Yu et al. [YZW+19] show how to construct a hash function from the LPN problem as follows:

Construction 2. *Let $S_w^m \subseteq \{0,1\}^m$ be the set of length-m vectors, where the domain is divided into w blocks of size m/w, and each block contains exactly a single 1. Let $\mathsf{LPNHash}_w^{n\times m}$ be the hash function family defined as follows: $h : S_w^m \to \{0,1\}^n$ is specified by a random matrix $A \in \mathbb{Z}_2^{n\times m}$. Then $h(x) = A \cdot x \bmod 2$.*

Remark 7. Brakerski et al. allow for a slightly more general domain where the inputs can have w 1's in any position. For our analysis of semi-regularity, however, it will be convenient to use the domain S_w^m as defined.

Theorem 3 ([BLVW19]). *Under the $(O(\log^2 n/n), \mathsf{poly}, \mathsf{poly})$-LPN assumption, $\mathsf{LPNHash}_w^{n\times m}$ is a PQ-CRHF for $m = \mathsf{poly}(n)$ and $w = O(n/\log n)$.*

Theorem 4 ([YZW+19]). *The following are true:*

- *Under the $(O(1), 2^{O(n^{0.5})}, 2^{O(n^{0.5+\epsilon})})$-LPN assumption, $\mathsf{LPNHash}_w^{n\times m}$ is a PQ-CRHF for $n = O(\log^2 \lambda)$, $m = \lambda$, and $w = O(\log^{1+2\epsilon} \lambda)$.*
- *Under the $(O(1), 2^{O(n/\log n)}, \mathsf{poly})$-LPN assumption, $\mathsf{LPNHash}_w^{n\times m}$ is a PQ-CRHF for $m = \mathsf{poly}(n)$ and $w = O(n/\log n)$.*
- *Under the $(O(n^{-0.5}), 2^{O(n^{0.5}/\log n)}, \mathsf{poly})$-LPN assumption, $\mathsf{LPNHash}_w^{n\times m}$ is a PQ-CRHF for $m = \mathsf{poly}(n)$ and $w = O(n/\log n)$.*

5.2 Semi-Regularity of LPN-Based Hashing

We now prove that $\mathsf{LPNHash}$ is semi-regular, for appropriate parameter choices.

Theorem 5. *For any m, n, w, let $\alpha := \sqrt{n(w/m)\ln 2}$. If $\alpha \le 1/2$ and $\alpha^w \le 2^{-n}$, then $\mathsf{LPNHash}_w^{n\times m}$ is semi-regular.*

Before proving Theorem 5, we observe an immediate corollary:

Corollary 2. *If LPN is hard in any of the parameter regimes in Theorems 3 or 4, then collapsing hash functions exist:*

Proof. By Theorem 2, it suffices to show that the settings of parameters in Theorems 3 and 4 satisfy the conditions of Theorem 5. For the settings where $m = \mathsf{poly}(n)$ and $w = O(n/\log n)$, we just need to set $m = n^c$ and $w = dn\log n$ where $cd \ge 2$. Then $\alpha = o(1)$ and

$$\alpha^w = \left(\frac{dn^2 \ln 2}{n^{1+c}\log n}\right)^{dn/2\log n} \le \left(\frac{1}{n^{c-1}}\right)^{dn/2\log n} \le 2^{-n} \ .$$

For the setting where $n = O(\log^2 \lambda), m = \lambda = 2^{n^{0.5}}, w = O(\log^{1+2\epsilon} \lambda)) = O(n^{0.5+\epsilon})$, we have $\alpha = \mathsf{poly}(n)2^{-O(n^{0.5})} \leq 2^{-O(n^{0.5-\epsilon/2})} = o(1)$ and $\alpha^w \leq 2^{-O(n^{1+\epsilon/2})} < 2^{-n}$. □

We now prove Theorem 5.

Proof. Our goal is to show that $\|h(S_w^m)\|_\infty = \mathsf{poly}/2^n$, which implies $H_\infty(h) \geq n - O(\log n)$. Since $H_2(h) \leq n$, this would establish semi-regularity.

We will write $A = (v_1, \ldots, v_m)$ for vectors $v_i \in \mathbb{Z}_2^n$. Let D_i be the distribution $v_{j_1} + v_{m/w+j_2} + \cdots v_{(m/w)(i-1)+j_i}$, where each j_i is uniform in $[m/w]$. Then $h(S_w^m) = D_w$.

Lemma 3. *Fix $v_1, \ldots, v_{(m/w)i}$. Suppose $\|D_i\|_\infty = f/2^n$. Then except with probability 2^{-n} over the choice of $v_{(m/w)i+1}, \ldots, v_{(m/w)(i+1)}$, $\|D_{i+1}\|_\infty \leq (1+g)/2^n$, where $g = f\sqrt{n(w/m)\ln 2}$*

Proof. For each x in $\{0,1\}^n$, define $p_x^{(i)} := \Pr[x \leftarrow D_i]$. Then

$$p_x^{(i+1)} = \frac{w}{m} \sum_{j=1}^{w/m} p^{(i)}_{x \oplus v_{(m/w)i+j}} \quad .$$

The $v_{(m/w)i+j}$ are just independent random vectors, so we can think of $p_x^{(i+1)}$ as a random variable which is the mean of w/m random samples of $p_{x'}^{(i)}$ for random x'. Each of the $p_{x'}^{(i)}$ are non-negative random variables with mean 2^{-n} (since they must sum to 1) and maximum $f \times 2^{-n}$. By Hoeffding's inequality,

$$\Pr[p_x^{(i+1)} > (1+g)/2^n] = \Pr[p_x^{(i+1)} - 2^{-n} > g/2^n] < e^{-2(m/w)\frac{g^2}{f^2}} \quad .$$

Union-bounding over all 2^n different x, we have that

$$\Pr[\|D_{i+1}\|_\infty > (1+g)/2^n] < 2^n \times e^{-2(m/w)\frac{g^2}{f^2}} \quad .$$

By setting $g = f\sqrt{n(w/m)\ln 2}$, the right-hand side becomes 2^{-n}, as desired. □

Notice that $\|D_0\|_\infty = 1$. Let $\alpha = \sqrt{n(w/m)\ln 2}$. Union-bounding over all $i = 1, \ldots, w$, we therefore have that

$$\|D_{i+1}\|_\infty \leq \alpha \|D_i\|_\infty + 2^{-n} \quad .$$

for all i. Then

$$\|D_w\|_\infty \leq \alpha^w \|D_0\|_\infty + \left(\sum_{i=0}^{w-1} \alpha^i\right) \times 2^{-n} \leq \alpha^w + \frac{1}{1-\alpha} \times 2^{-n} \quad .$$

If we set α so that $\alpha^w \leq 2^{-n}$ and $\alpha \leq 1/2$, we have that $\|D_w\|_\infty \leq 3 \times 2^{-n}$, showing that LPNHash is semi-regular. □

6 Collapsing Hashes from Expanders

Charles, Goren, and Lauter [CLG09], abstracting earlier ideas of Tillich and Zémor [TZ94], propose an elegant way to construct collision resistant hash functions from exponentially-large expander graphs, whose collision-resistance follows from the assumed difficulty of finding cycles in the graphs. A number of graphs have been proposed for use in hash functions, such as:

- Charles et al. [CLG09] propose using isogeny graph on certain elliptic curves.
- Fuchs et al. [FLLT21] propose using the graph of Markov Triples.

We show that expander-based hashes satisfy our regularity condition, and hence we can obtain collapsing hash functions under the same computational assumptions on expanders as for collision resistance.

6.1 Expander Graphs

Let $G = (V, E)$ be an undirected graph. G is d-regular if every $v \in V$ has exactly d neighbors. Throughout, we will always assume our graphs are regular. Let $A = A(G)$ denote the adjacency matrix of G: the $|V| \times |V|$ matrix such that $A_{i,j}$ if $(i, j) \in E$ and 0 otherwise. Since A is symmetric, it has $|V|$ real eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. For a d-regular graph, $\lambda_1 = d$.

There are several equivalent definitions of expander graphs; the following linear-algebraic definition captures the only property we will need.

Definition 5. *A connected d-regular graph G is a $(|V|, d, \delta)$-expander graph if $\lambda_2 \leq \delta d$.*

Walks on Expanders. Let G be a d-regular graph, and let $v_0 \in V$ be a node. A walk on G starting from v_0 is simply a sequence $(v_0, v_1, v_2, \dots)$ such that $(v_{i-1}, v_i) \in E$ for all $i > 0$. A *random walk* is one where v_{i+1} is chosen uniformly from the set of neighbors of v_i. A *non-backtracking* walk is one where $v_{i-1} \neq v_{i+1}$ for all $i > 0$, and a *random non-backtracking* walk is a walk where v_{i+1} is chosen uniformly from the neighbors of v_i other than v_{i-1}.

For a d regular graph, the nodes v_i for a random walk and random non-backtracking walk will converge to the uniform distribution over V as $i \to \infty$. We will use the notion of *mixing* time to characterize how fast this occurs.

Definition 6. *The* mixing time *of a random walk starting at v_0 is defined as*

$$\tau(G) = \min_t \left\{ \left| \Pr[v_t = u] - \frac{1}{|V|} \right| \leq \frac{1}{2|V|} \forall u \in V \right\} \ ,$$

where $\Pr[v_t = u]$ is the probability that $v_t = u$ in the walk. The mixing time for a random non-backtracking walk is defined as $\tilde{\tau}(G)$, and is defined analogously.

For both backtracking and non-backtracking walks, the mixing time is at most $O(\log(|V|)/(1 - \delta))$. The backtracking case has long been known, and the non-backtracking case follows from the fact that non-backtracking walks mix at least as fast, as shown by [ABLS07].

6.2 Hash Functions Based on Expanders

Let $\mathcal{G} = (\mathcal{G}_\lambda)_\lambda$ where each $\mathcal{G}_\lambda$ is a family of d-regular connected graphs where each $G = (V, E) \in \mathcal{G}_\lambda$ is exponentially large and *implicitly represented*. That is, $V \subseteq \{0,1\}^{n(\lambda)}$, and each G is represented by a polynomial-size string $\mathsf{Desc}(G)$. There is an efficient procedure which computes the neighbors of any $v \in V$, given $\mathsf{Desc}(G)$. We assume that $\mathsf{Desc}(G)$ includes a distinguished node v_0, and that it is possible to efficiently sample $\mathsf{Desc}(G)$ for a random $G \leftarrow \mathcal{G}_\lambda$.

Definition 7. *The* Cycle Finding *problem is hard in $\mathcal{G}$ if, for any QPT $\mathcal{A}$, $\mathcal{A}(\mathsf{Desc}(G)), G \leftarrow \mathcal{G}_\lambda$ outputs a simple cycle in G with negligible probability.*

Based on cycle finding hardness, [CLG09] constructs the following hash:

Construction 3 ([CLG09]). *Let $\mathsf{ExHash}_\mathcal{G}$ be the distribution over functions $h_{\mathsf{Desc}(G)} : [d-1]^t \to \{0,1\}^{n(\lambda)}$ for a random $G \leftarrow \mathcal{G}_\lambda$ defined as follows: interpret each element x of $[d-1]^n$ as a length-n non-backtracking walk in G starting from v_0. That is, on the ith step, if the walk is currently at node v_i and was previously at v_{i-1}, then x_i selects amongst the $d-1$ neighbors of v_i other than v_{i-1}. That neighbor will be v_{i+1}. Let v_t be the end of the walk. Then $h_{\mathsf{Desc}(G)}(x) = v_t$.*

Theorem 6 ([CLG09]). *$\mathsf{ExHash}_\mathcal{G}$ is a PQ-CRHF if cycle finding is hard in $\mathcal{G}$.*

Proof. We give the proof for completeness. Any collision in $h_{\mathsf{Desc}(G)}$ gives two non-backtracking walks $W_0 \neq W_1$ that start at v_0 and end at the same node v. Assume without loss of generality that the nodes immediately before v in W_1, W_1 are different. Let v_1 be the last node before v where the walks coincide. Then by concatenating the two paths from v_1 to v under W_0, W_1 gives a simple cycle. □

[CLG09] propose using expander graphs as a minimal criteria for selecting $\mathcal{G}$ where the cycle finding problem is hard. A uniformly random input to $\mathsf{ExHash}_\mathcal{G}$ corresponds to a random non-backtracking walk on $\mathcal{G}$. Since the mixing time of an expander is logarithmic in $|V|$, it is polynomial for implicitly represented graphs. Once the walk mixes, no node in the graph is more likely than $2/|V|$, implying $\|h\|_\infty \leq 2/|V|$. Meanwhile, $\|h\|_2^2 \geq 1/|V|$. Therefore, for a polynomial-length input, $\mathsf{ExHash}_\mathcal{G}$ is semi-regular with $r \leq 2$. Therefore, we have the following:

Corollary 3. *Suppose $\mathcal{G}$ is a family of $(|V_\lambda|, d, \delta)$-expander graph for a constant δ. Then if cycle finding is hard for $\mathcal{G}$, there exists collapsing hash functions.*

When V is an appropriate set of elliptic curves and E are isogenies as proposed by [CLG09], cycle-finding is a well-known challenging problem. The graph of Markov triples has been explored by [FLLT21]. Other instantiations have been proposed [CLG09, TZ94, PLQ12], but they have weaknesses [PLQ08].

7 Toward Collapsing Hashes from General Collision Resistance

Here, we discuss the possibility of obtaining collapsing hashes from more general PQ-CRHFs. In particular, we are interested in the case of symmetric hash

functions such as SHA2 or SHA3. It seems plausible that SHA2 or SHA3 would be semi-regular: after all, if a hash function had certain images that were far more likely than others, this would be considered a significant design weakness. Unfortunately, we do not know how to prove unconditionally that, say, SHA2 or SHA3 are semi-regular. Instead, we simply conjecture it. The following shows that this assumption is justified in the random oracle mode:

Lemma 4. *Random oracles are semi-regular. In particular, for λ bit outputs, a compressing random oracle has regularity at most λ.*

Proof. By a standard balls-and-bins argument, for a random function $F : \{0,1\}^m \to \{0,1\}^\lambda$, the most likely output has probability $H_\infty(F) \leq O(\lambda 2^{-\lambda})$, with all but negligible probability. On the other hand, $\|F\|_2^2 \geq 2^{-\lambda}$. Thus F has regularity at most $O(\lambda)$. □

Since SHA2 or SHA3 are often modeled as random oracles, it therefore seems reasonable to conjecture that they are semi-regular. Note that this is potentially very different than assuming SHA2 or SHA3 are collapsing, even though random oracles are collapsing. Indeed, the analysis of SHA2 and SHA3 has usually focused on classical security properties. Semi-regularity is a simple classical property, whereas collapsing is a more complicated *inherently quantum* property. Under the assumed quantum collision resistance and assumed regularity of either SHA2 or SHA3, we therefore obtain a standard-model collapsing hash function from classically-defined properties, which are much better understood.

7.1 Collapsing from Optimal Collision Resistance

Here, we give another, simpler, approach for justifying building collapsing hashes from SHA2 or SHA3. Namely, we observe that symmetric hash functions are usually treated as having optimal collision resistance, defined as follows:

Definition 8 (Optimal Collision Resistance). *$\mathcal{H}$ is a post-quantum optimally collision resistant if, for every QPT algorithm $\mathcal{A}$, there exists a a polynomial $q(\lambda)$ such that*

$$\Pr\left[\begin{smallmatrix} x_0 \neq x_1, \text{ and} \\ h(x_0)=h(x_1) \end{smallmatrix} : \begin{smallmatrix} h\leftarrow\mathcal{H}_\lambda \\ (x_0,x_1)\leftarrow\mathcal{A}(h) \end{smallmatrix}\right] < \frac{q(\lambda)}{|\mathcal{Y}_\lambda|} \ ,$$

where $\mathcal{Y}_\lambda$ is the co-domain of h.

If SHA2 or SHA3 turned out to not be optimally collision resistant, this would be considered a major weakness of the functions. It is therefore plausible to conjecture such hardness.

Theorem 7. *Suppose $\mathcal{H}$ is a hash function with domain $\mathcal{X}_\lambda$ and co-domain $\mathcal{Y}_\lambda$ such that $|\mathcal{X}_\lambda|/|\mathcal{Y}|_\lambda$ is polynomial. Equivalently, suppose $\mathcal{H}$ compressed by at most logarithmically many bits. Then if $\mathcal{H}$ is post-quantum optimally collision resistant, it is also collapsing.*

Note that the function $\mathcal{H}$ may be optimally collision resistant, but fail to be semi-regular: for example there may be a single input that is very likely, but infeasible to find a pre-image of in polynomial time. Such an $\mathcal{H}$ is not semi-regular, but could plausibly be optimally collision resistant. Thus, Theorem 7 offers a distinct alternative to assuming semi-regularity, trading off a structural assumption for a stronger hardness assumptions. Depending on the analysis performed, either approach may be preferred.

Proof. Let $\mathcal{A}$ be a collapsing adversary with non-negligible advantage ϵ. Let $|\psi\rangle$ be the superposition of inputs to h produced by $\mathcal{A}$, and y be the measured image of $|\psi\rangle$. We give a simple adversary $\mathcal{B}$ for optimal collision resistance. $\mathcal{B}$ first runs $\mathcal{A}(h)$ to get $|\psi\rangle$, and then applies the measurement $\mathcal{M}^h_{\mathcal{Y}}$ to get y. Then it simply measures $|\psi\rangle$ to get a pre-image x_0 such that $h(x_0) = y$. It finally chooses a uniformly random input $x_1 \in \mathcal{X}_\lambda$ and outputs (x_0, x_1).

By the optimal collision resistance of $\mathcal{H}$, we know that $\mathcal{B}$ finds a collision with probability at most $p/|\mathcal{Y}_\lambda|$ for a polynomial p. But the probability $\mathcal{B}$ finds a collision is just the expected fraction of $\mathcal{X}_\lambda$ that are pre-images of y but not equal to x_0. Since $\mathcal{X}_\lambda$ is only polynomially larger than $\mathcal{Y}_\lambda$, we therefore have that the expected number of pre-images of y is polynomial ℓ. In particular, with probability at least $1/2$, the number of pre-images is at most 2ℓ.

But now we can use Lemma 1 to construct a different collision finding adversary $\mathcal{C}$. This is basically identical to the proof of Theorem 1: if y has ℓ pre-images, then $\mathcal{C}$ finds a collision with probability at least $2\epsilon^2/(\ell - 1)$. Therefore, $\mathcal{C}$ finds a collision with probability at least $\epsilon^2/2\ell$, which is non-negligible and in particular violates the optimal security of $\mathcal{H}$. □

References

ABLS07. Noga Alon, Itai Benjamini, Eyal Lubetzky, and Sasha Sodin. Non-backtracking random walks mix faster. *Communications in Contemporary Mathematics*, 9:585–603, 2007.

AGKZ20. Ryan Amos, Marios Georgiou, Aggelos Kiayias, and Mark Zhandry. One-shot signatures and applications to hybrid quantum/classical authentication. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *52nd ACM STOC*, pages 255–268. ACM Press, June 2020.

AMG+16. Matthew Amy, Olivia Di Matteo, Vlad Gheorghiu, Michele Mosca, Alex Parent, and John M. Schanck. Estimating the cost of generic quantum pre-image attacks on SHA-2 and SHA-3. In Roberto Avanzi and Howard M. Heys, editors, *SAC 2016*, volume 10532 of *LNCS*, pages 317–337. Springer, Heidelberg, August 2016.

AMRS20. Gorjan Alagic, Christian Majenz, Alexander Russell, and Fang Song. Quantum-access-secure message authentication via blind-unforgeability. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 788–817. Springer, Heidelberg, May 2020.

ARU14. Andris Ambainis, Ansis Rosmanis, and Dominique Unruh. Quantum attacks on classical proof systems: The hardness of quantum rewinding. In *55th FOCS*, pages 474–483. IEEE Computer Society Press, October 2014.

BHT97. Gilles Brassard, Peter Høyer, and Alain Tapp. Quantum Algorithm for the Collision Problem. *ACM SIGACT News (Cryptology Column)*, 28:14–19, 1997.

BLVW19. Zvika Brakerski, Vadim Lyubashevsky, Vinod Vaikuntanathan, and Daniel Wichs. Worst-case hardness for LPN and cryptographic hashing via code smoothing. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part III*, volume 11478 of *LNCS*, pages 619–635. Springer, Heidelberg, May 2019.

CCY21. Nai-Hui Chia, Kai-Min Chung, and Takashi Yamakawa. A black-box approach to post-quantum zero-knowledge in constant rounds. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 315–345, Virtual Event, August 2021. Springer, Heidelberg.

CHS19. Jan Czajkowski, Andreas Hülsing, and Christian Schaffner. Quantum indistinguishability of random sponges. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 296–325. Springer, Heidelberg, August 2019.

CLG09. Denis Xavier Charles, Kristin E. Lauter, and Eyal Z. Goren. Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22(1):93–113, January 2009.

CMSZ21. Alessandro Chiesa, Fermi Ma, Nicholas Spooner, and Mark Zhandry. Post-quantum succinct arguments. In *Proceedings of FOCS 2021*, 2021.

Cou06. Jean-Marc Couveignes. Hard homogeneous spaces. Cryptology ePrint Archive, Report 2006/291, 2006. https://eprint.iacr.org/2006/291.

CX22. Shujiao Cao and Rui Xue. The gap is sensitive to size of preimages: Collapsing property doesn't go beyond quantum collision-resistance for preimages bounded hash functions. In *CRYPTO 2022 (to appear)*, 2022.

Cza21. Jan Czajkowski. Quantum indifferentiability of SHA-3. Cryptology ePrint Archive, Report 2021/192, 2021. https://eprint.iacr.org/2021/192.

DFMS19. Jelle Don, Serge Fehr, Christian Majenz, and Christian Schaffner. Security of the Fiat-Shamir transformation in the quantum random-oracle model. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 356–383. Springer, Heidelberg, August 2019.

FLLT21. Elena Fuchs, Kristin Lauter, Matthew Litman, and Austin Tran. A cryptographic hash function from markoff triples. Cryptology ePrint Archive, Report 2021/983, 2021. https://eprint.iacr.org/2021/983.

GKL88. Oded Goldreich, Hugo Krawczyk, and Michael Luby. On the existence of pseudorandom generators (extended abstract). In *29th FOCS*, pages 12–24. IEEE Computer Society Press, October 1988.

HILL99. Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

HS21. Akinori Hosoyamada and Yu Sasaki. Quantum collision attacks on reduced SHA-256 and SHA-512. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 616–646, Virtual Event, August 2021. Springer, Heidelberg.

LMS21. Alex Lombardi, Fermi Ma, and Nicholas Spooner. Post-quantum zero knowledge, revisited (or: How to do quantum rewinding undetectably). Cryptology ePrint Archive, Report 2021/1543, 2021. https://eprint.iacr.org/2021/1543.

LZ19. Qipeng Liu and Mark Zhandry. Revisiting post-quantum Fiat-Shamir. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 326–355. Springer, Heidelberg, August 2019.

PLQ08. Christophe Petit, Kristin Lauter, and Jean-Jacques Quisquater. Full cryptanalysis of LPS and Morgenstern hash functions. In Rafail Ostrovsky, Roberto De Prisco, and Ivan Visconti, editors, *SCN 08*, volume 5229 of *LNCS*, pages 263–277. Springer, Heidelberg, September 2008.

PLQ12. Christophe Petit, Kristin E. Lauter, and Jean-Jacques Quisquater. Cayley hashes: A class of efficient graph-based hash functions. 2012.

Reg05. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th ACM STOC*, pages 84–93. ACM Press, May 2005.

Rob21. Bhaskar Roberts. Security analysis of quantum lightning. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part II*, volume 12697 of *LNCS*, pages 562–567. Springer, Heidelberg, October 2021.

RS06. Alexander Rostovtsev and Anton Stolbunov. Public-Key Cryptosystem Based On Isogenies. Cryptology ePrint Archive, Report 2006/145, 2006. https://eprint.iacr.org/2006/145.

Sho94. Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th FOCS*, pages 124–134. IEEE Computer Society Press, November 1994.

TZ94. Jean-Pierre Tillich and Gilles Zémor. Hashing with SL_2. In Yvo Desmedt, editor, *CRYPTO'94*, volume 839 of *LNCS*, pages 40–49. Springer, Heidelberg, August 1994.

Unr16a. Dominique Unruh. Collapse-binding quantum commitments without random oracles. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part II*, volume 10032 of *LNCS*, pages 166–195. Springer, Heidelberg, December 2016.

Unr16b. Dominique Unruh. Computationally binding quantum commitments. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 497–527. Springer, Heidelberg, May 2016.

Unr17. Dominique Unruh. Collapsing sponges: Post-quantum security of the sponge construction. Cryptology ePrint Archive, Report 2017/282, 2017. https://eprint.iacr.org/2017/282.

VDG98. Jeroen Van De Graaf. *Towards a Formal Definition of Security for Quantum Protocols*. PhD thesis, CAN, 1998. AAINQ35648.

Wat06. John Watrous. Zero-knowledge against quantum attacks. In Jon M. Kleinberg, editor, *38th ACM STOC*, pages 296–305. ACM Press, May 2006.

YZW+19. Yu Yu, Jiang Zhang, Jian Weng, Chun Guo, and Xiangxue Li. Collision resistant hashing from sub-exponential learning parity with noise. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT 2019, Part II*, volume 11922 of *LNCS*, pages 3–24. Springer, Heidelberg, December 2019.

Zha19a. Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 239–268. Springer, Heidelberg, August 2019.

Zha19b. Mark Zhandry. Quantum lightning never strikes the same state twice. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part III*, volume 11478 of *LNCS*, pages 408–438. Springer, Heidelberg, May 2019.