

Bitwise-Optimal Cryptography: From One-Wayness to Pseudorandomness and Target Collision Resistance*

Benny Applebaum
Tel-Aviv University
Tel-Aviv, Israel
benny.applebaum@gmail.com

October 4, 2026

Abstract

We study cryptographic primitives that are both locally computable (i.e., in NC^0) and exponentially secure. For pseudorandom generators (PRGs) and universal one-way hash functions (UOWHFs), we further require linear stretch and linear compression, respectively, which is essentially the best one can hope for in this setting. Such primitives simultaneously achieve an extreme level of security and efficiency: each output bit inspects only a constant number of input bits, while each input bit buys a constant amount of security and expansion/shrinkage, in an amortized sense. We refer to such primitives as *bitwise optimal*.

We prove that bitwise-optimal PRGs and UOWHFs can be obtained from *any* exponentially secure one-way function (OWF) in NC^0 , thereby establishing an equivalence between these three primitives in the bitwise-optimal regime. Notably, an analogous equivalence is not known in the exponential-security regime for general, unrestricted cryptographic primitives without imposing an additional regularity condition.

Our results combine the machinery of the author (Applebaum, FOCS'17) with new structural results on locally computable functions that may be of independent interest. We prove a sparsification theorem that reduces the output length of any exponentially secure local OWF to $O(n)$ while preserving exponential hardness, an input-locality reduction that bounds the number of outputs affected by each input bit, and a structural theorem showing that functions with bounded input locality are typically almost regular. Together, these ingredients remove the regularity assumption required by previous constructions and yield a non-black-box transformation from exponentially secure OWFs in NC^0 to bitwise-optimal PRGs and UOWHFs. As an additional contribution, we prove a projection-based compression theorem for locally samplable sources.

1 Introduction

The study of “cryptography with low complexity” seeks to reconcile two seemingly conflicting goals: achieving strong security while minimizing computational complexity. This combination of “hardness” and “simplicity” is motivated both algorithmically—by the goal of minimizing the cost of cryptography—and from a complexity-theoretic perspective—by the goal of understanding the intractability of simple functions. In this paper, we push this tension to the extreme by considering cryptographic primitives $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$ that are simultaneously *exponentially secure* and *locally computable*, focusing on three basic primitives: one-way functions (OWFs), pseudorandom generators (PRGs), and universal one-way hash functions (UOWHFs), also known as target-collision-resistant (TCR) hash functions.

*This research is supported by Israel Science Foundation (ISF) grant no. 2805/21 and by the European Union (ERC-2022-ADG) under grant agreement no.101097959 NFITSC.

Informally, exponential security means that, for some constant $\beta > 0$, adversaries of size $2^{\beta n}$ succeed with probability (or distinguishing advantage) at most $2^{-\beta n}$, while locality means that each output bit depends on only $d = O(1)$ input bits. For PRGs and UOWHFs, whose usefulness crucially depends on their length expansion or compression, we additionally require *linear stretch* and *linear compression*, respectively: a PRG should expand n bits to at least cn bits for some constant $c > 1$, while a UOWHF should compress n bits to at most cn bits for some constant $c < 1$. Up to constant factors, this is essentially the best one can hope for while maintaining locality and exponential security. (For UOWHFs this is immediate, for PRGs this follows from a combinatorial argument of [8, Corollary 6.4].)

Put differently, each output bit should inspect only a constant number of input bits, while each input bit should buy a constant amount of security and expansion/shrinkage in an amortized sense. We therefore refer to such primitives as *bitwise-optimal*. For keyed primitives, we additionally require a public key of length $O(n)$ and constant locality jointly in the message and key bits.

This notion simultaneously strengthens several previous notions of low complexity cryptography, including cryptography in NC^0 with polynomial hardness and linear-size cryptography with polynomial or exponential hardness (e.g., [7, 33, 15] and their follow-ups). Indeed, every NC^0 function from $\{0, 1\}^n$ to $\{0, 1\}^{\Theta(n)}$ admits a linear-size circuit. See Section 1.2 for further discussion. Throughout, locality refers to *output locality*; jumping ahead, bounded *input locality*—the number of output bits affected by a given input bit—will also play an important role in our proofs.

Our main result shows that bitwise-optimal one-way functions can be transformed into bitwise-optimal PRGs and bitwise-optimal UOWHFs.

Theorem 1.1 (Main theorem). *Assuming the existence of an exponentially hard one-way function in NC^0 , there exist an exponentially secure linear-stretch pseudorandom generator in NC^0 and an exponentially secure linear-compression universal one-way hash function in NC^0 . In fact, both primitives can be realized with locality four. For the UOWHF, the public key has length $O(n)$, and locality four holds jointly in the message and key bits; namely, the evaluation map $(x, k) \mapsto h_k(x)$ is four-local.*

Remark 1.2 (Locality vs. stretch/compression). *We mention that 4-local PRGs cannot have super-linear stretch, even if one only requires polynomial hardness [37]. On the other hand, given an exponentially secure PRG (resp., UOWHF) of constant locality and linear stretch (resp., linear compression), say with a constant expansion (resp., compression) factor, one can obtain any desired constant stretch $C' > 1$ (resp., compression factor $C' < 1$), while preserving exponential security. This can be achieved, for example, by constant-depth tree-like composition, at the expense of increasing the locality to a larger constant.*

For PRGs, a previous construction of bitwise-optimal generators relied on bitwise-optimal OWFs satisfying an additional almost-regularity condition [4]. Theorem 1.1 removes this extra requirement and bases such generators on arbitrary exponentially hard local OWFs. For UOWHFs, the improvement is stronger: to the best of our knowledge, our construction is the first bitwise-optimal UOWHF whose security can be based on a one-wayness assumption alone; previous constructions required stronger assumptions tailored to target-collision resistance [12, 13, 5]. Thus, in both cases, Theorem 1.1 brings the assumption down to bitwise-optimal one-wayness. Indeed, since bitwise-optimal PRGs and UOWHFs immediately imply bitwise-optimal OWFs, Theorem 1.1 establishes equivalence between the three primitives. (See Proposition 2.4.)

Notably, an analogous equivalence is not known in the exponential-security regime for general, unrestricted cryptographic primitives. At the level of polynomial security, arbitrary OWFs are known to imply both PRGs [29, 28] and UOWHFs [39, 34, 27]. However, the known general transformations incur a super-linear blowup in the relevant security parameter and therefore do not preserve exponential security as a function of the input length of the resulting primitive. In particular, even without imposing locality or insisting on linear stretch/compression, it is not known how to transform an arbitrary exponentially hard OWF into a PRG or UOWHF that remains exponentially secure with respect to its own input length. Such transformations

are known when the underlying OWF satisfies additional structural conditions, most notably some form of *regularity*: roughly speaking, all strings y in the image of f have approximately the same fiber size

$$|f^{-1}(y)| := |\{x : f(x) = y\}|.$$

We overcome this *regularity barrier* by showing that exponentially hard locally computable OWFs can be transformed into functions with a sufficient degree of regularity. In this sense, approximate regularity comes “for free” in the local setting. Thus, somewhat surprisingly, the desired equivalence is easier to establish for local primitives than for unrestricted ones. This provides an interesting, and relatively rare, example in which computational simplicity of the underlying primitive enables a stronger cryptographic transformation. Indeed, unlike previous NC^0 transformations between cryptographic primitives (e.g., [7, 6, 4]), whose security does not rely on the locality (or even efficiency) of the underlying primitive, the proof of Theorem 1.1 exploits locality itself in order to establish security.¹

1.1 Technical overview

Our starting point is the work of Applebaum [4], which, building on [33, 15], develops locally computable and randomness-efficient analogues of pairwise-independent hashing and related information-theoretic tools, such as list-decodable codes. As shown in [4], this machinery can be used to transform an exponentially hard local OWF $f : \{0, 1\}^n \rightarrow \{0, 1\}^{O(n)}$ that is *almost regular* into an exponentially hard local PRG with linear stretch. Here, almost regularity means that, for some ℓ , the fiber sizes

$$|f^{-1}(y)|, \quad y \in \text{Im}(f),$$

all lie between 2^ℓ and $2^{\ell+\alpha n}$, for some constant $\alpha > 0$. For the reduction to preserve exponential security, the regularity deviation α must be sufficiently small compared to the hardness parameter β .

We show that, by adapting existing constructions, the same starting point also suffices for constructing exponentially secure local UOWHFs with linear compression. Thus, the main technical task is to show that an arbitrary exponentially hard local OWF can be transformed into one that is sufficiently close to almost regular while preserving exponential hardness and locality. This will be done in several steps. For ease of presentation, we start from the end.

1.1.1 Input locality implies typical almost-regularity

Suppose that our one-way function $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$ has constant *input locality* [9], in the sense that each input bit affects at most $D = O(1)$ output bits. We claim that such a function must be “close” to almost regular.

The idea is to start with a suitably chosen set $T \subseteq \text{Im}(f)$ of outputs and consider its preimage $A = f^{-1}(T)$. By isoperimetry, a small Hamming neighborhood A_t of A already contains almost all inputs. Bounded input locality then implies that $f(A_t)$ is contained in a small Hamming neighborhood of T , and therefore cannot contain too many outputs whose probability is much smaller than those in T . On the other hand, by the choice of T , outputs of significantly larger probability carry only negligible total mass. Together, these two bounds confine almost all of the output probability mass to a narrow probability window. Formally, for every constant $\alpha > 0$, there exists a constant $c = c(\alpha, D) > 0$ such that, except for a 2^{-cn} fraction of the inputs, the value

$$\log |f^{-1}(f(x))|$$

lies in an interval of width αn . We refer to this property as *typical almost regularity*.

¹Another related example, discussed later, is the Bogdanov–Rosen [17] hardness-amplification theorem for functions with constant input locality.

Fortunately, the aforementioned transformations from almost-regular bitwise-optimal OWFs to bitwise-optimal PRGs and UOWHFs can be extended, with some care, to this weaker notion of typical almost regularity.

Remark 1.3. *Interestingly, the same isoperimetric, or “blowing-up,” phenomenon for the Hamming cube was used by Bogdanov and Rosen [17] in a rather different context, namely hardness amplification for functions with bounded input locality. Roughly speaking, they use blowing-up to convert inversion on a non-negligible set of inputs into inversion on almost all inputs by combining the inverter with a local search procedure. Our use is different and purely structural: bounded input locality alone forces concentration of the output probabilities, independently of any cryptographic hardness assumption.*

1.1.2 From linear output length to bounded input locality

We next explain how to obtain bounded input locality. Output locality by itself does not imply input locality: a single input bit may participate in arbitrarily many local output functions. Concretely, in the regime of polynomial hardness, generic transformations from NC^1 primitives yield cryptographic constructions with constant *output locality* [7], but do not in general provide constant *input locality*.² The picture changes, however, when constant output locality is combined with *linear output length* and exponential hardness.

Indeed, if a function f with output locality $d = O(1)$ has $m = O(n)$ outputs, then the average input locality is at most

$$\bar{D} = md/n = O(1).$$

By an averaging argument, if we fix a γ -fraction of the inputs with highest input locality, the resulting restricted function has input locality at most $\bar{D}/\gamma = O(1)$. By choosing γ sufficiently smaller than the hardness parameter β , exponential hardness is preserved. We conclude that it suffices to *sparsify* the given one-way function f , reducing its number of outputs to $m = O(n)$ while preserving exponential hardness.³ Indeed this last step forms our main structural result.

1.1.3 An ideal sparsification theorem

Our goal is to show that any exponentially hard local one-way function $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$ with $m = \text{poly}(n)$ can be converted into an exponentially hard local one-way function $g : \{0, 1\}^n \rightarrow \{0, 1\}^{m'}$ with a linear number of outputs $m' = O(n)$. A natural approach is to take g to be an output-restricted version of f . Namely, find an m' -subset $S \subseteq [m]$ of the outputs of f , and define

$$g(x) = f(x)_S := (f_i(x))_{i \in S}.$$

To prove one-wayness, it suffices to show that $g(x)$ preserves all the information contained in $f(x)$. That is, for every pair x, x' ,

$$f(x) = f(x') \iff g(x) = g(x').$$

Equivalently, there is a (possibly inefficient) *reconstruction* procedure that, given $g(x)$, recovers $f(x)$.

²Indeed, constant input locality is a substantially stronger requirement. For some primitives, such as signatures and MACs, it is known to be impossible, whereas for others, such as OWFs and PRGs, constructions with constant input locality are known only under a significantly more restricted class of assumptions [9].

³This is analogous to the celebrated sparsification lemma of Impagliazzo, Paturi, and Zane [32], which, in our terminology, shows that if f is exponentially hard to invert in the worst case (for some output y), then there exists a locally computable function g with $O(n)$ outputs that remains exponentially hard to invert in the worst case. Unfortunately, it is not clear how to adapt their techniques to obtain the type of sparsification needed in the cryptographic setting. See [4] for further discussion of the relation between exponentially strong local cryptography and worst-case exponential hardness assumptions such as ETH and Gap-ETH.

At first sight, the existence of such a restriction seems plausible from the following information-theoretic intuition. Let X be uniform over $\{0, 1\}^n$, and let Y_i denote the random variable corresponding to the i th output of f . Constant locality guarantees that every nonconstant output carries a constant amount of information about X . One may therefore hope to iteratively add to S the most informative output, conditioned on those already collected. Since X contains only n bits of information, if each step contributed a constant amount of new information, the process would terminate after at most $O(n)$ steps.

Unfortunately, this approach does not yield the desired strong sparsification result. In fact, strong sparsification, as defined above, is impossible in general. To see this, consider the 2-local *pairwise-product* function $f(x)$ that outputs all $\binom{n}{2}$ degree-2 monomials $(x_i x_j)_{i < j}$. Even if we retain all but a single output, say $x_1 x_2$, reconstruction fails. Indeed, the strings 0^n and 110^{n-2} have the same (all-zero) image under g , but different images under f .

This suggests aiming for a more modest sparsification theorem, in which reconstruction holds for “almost all” inputs. The information-theoretic intuition above would support such a result if one could show that, as long as the remaining output Y contains $\Omega(n)$ bits of information, there is some output Y_i that contributes a constant amount of new information conditioned on the outputs already collected. It turns out that establishing such a statement is trickier than it may first appear.

1.1.4 Our sparsification theorem

Our actual sparsification theorem provides a very strong form of reconstruction after fixing a small fraction of the input coordinates. Specifically, we show that for every constant $\rho > 0$, after applying a *specially crafted* restriction R that fixes at most a ρ -fraction of the inputs of a d -local function $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$, one can retain a linear number $O_{d,\rho}(n)$ of outputs $S \subseteq [m]$ that suffice to reconstruct the entire restricted output $f(R[x])$ from

$$g(R[x]) := f(R[x])_S.$$

Moreover, the reconstruction can be performed efficiently by a local mapping D . Put differently, the R -restricted function admits the decomposition

$$f(R[x]) = D(f(R[x])_S).$$

Thus, after a small input restriction, the full local map can be faithfully represented by only linearly many of its original outputs.

The theorem is purely combinatorial and makes no one-wayness assumption. Cryptography enters only through the choice of the restriction parameter ρ : if f is exponentially hard, then taking ρ sufficiently small preserves exponential one-wayness, while the above reconstruction property guarantees that projecting to the retained outputs does not introduce new collisions that could facilitate inversion. We therefore obtain an exponentially hard local OWF with only $O(n)$ outputs.

At a high level, the proof is based on the following procedure. Start with a single input variable x_i and distinguish between the following three cases. If many outputs depend on x_i but otherwise use disjoint variables, a small random restriction has many independent chances to turn one of them into x_i or its negation. Retaining such an output then reveals x_i . If only constantly many outputs depend on x_i , we can simply retain them all. The problematic case is when x_i occurs in many outputs, but too few of these occurrences are disjoint outside x_i . In this case, every occurrence involves at least one additional variable from a constant-size list. We therefore enlarge the singleton $\{i\}$ to the pairs formed with these variables, and examine information about these pairs rather than insisting on recovering x_i alone.

For each pair, the relevant pieces of information are the Boolean functions obtained by fixing the remaining inputs of an output—for example, the AND of the two variables. Each piece again leads to the same three cases: many occurrences with disjoint surrounding variables allow exposure by a random restriction; few

occurrences can be retained directly; and many overlapping occurrences lead to constantly many larger cores, now triples. Starting from the n singletons, this process collects only $O(n)$ core sets: each step has bounded branching, and no core needs more than d variables. The supports of directly retained outputs are also included. After applying the random restriction and fixing a few additional variables to repair failed exposures, we retain one representative of every distinct nonconstant restricted output function supported on one of these cores. Each core supports only constantly many Boolean functions, so only $O(n)$ outputs are retained. The main combinatorial argument shows that these outputs determine the entire restricted map. Finally, the restriction is derandomized by conditional expectations.

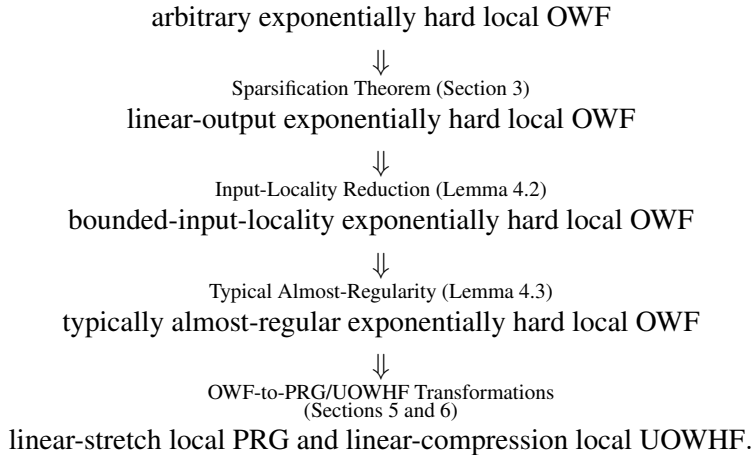
It is instructive to revisit the pairwise-product map. In this case, once we fix a single variable to one, say $x_1 = 1$, it suffices to retain the $n - 1$ outputs $(x_1 x_i)_{i \neq 1}$, since they completely reveal the remaining inputs $x_2, \dots, x_n$, and hence all the other outputs can be reconstructed from them.

Remark 1.4 (An alternative perspective: Projection-based compression for local sources). *Taking X to be uniform over $\{0, 1\}^n$, the output distribution $Y = f(X)$ is a locally samplable distribution, also called a local source [43]. Local sources have been studied extensively in the context of the complexity of sampling, randomness extraction, and sampling lower bounds [43, 21, 44, 45, 1, 46, 31].*

Our sparsification theorem provides a complementary structural perspective: a projection-based compression theorem for local sources. A refinement of its proof shows that, for every constant loss parameter $0 < \epsilon < 1$, there is a fixing of at most ϵn bits of the underlying randomness such that the resulting source has Shannon entropy at least $H(Y) - \epsilon n$ and admits exact compression to $O_{d,\epsilon}(n)$ of its own coordinates. Moreover, the entire restricted source can be reconstructed from this projection by a constant-local map. See Corollary 3.3 for details.

1.1.5 Putting it all together

Combining the steps gives the following chain:



1.2 Related work

From OWFs to PRGs. Early constructions of pseudorandom generators were based on one-way permutations [16, 47]. Goldreich, Krawczyk, and Luby [24] extended this to regular one-way functions, and the seminal result of Håstad, Impagliazzo, Levin, and Luby [29] established that arbitrary one-way functions suffice for constructing PRGs. Subsequent work simplified the construction and improved its quantitative efficiency [28, 42]. Nevertheless, the known general transformations incur superlinear parameter losses and do not transform an exponentially hard OWF into a PRG that is exponentially secure as a function

of its own seed length. A central source of this loss is the need to “flatten” the output distribution of an arbitrary OWF before extracting pseudorandomness. Chen, Göös, Vadhan, and Zhang [20] showed that any black-box entropy-flattening procedure requires $\Omega(n^2)$ oracle calls, matching the complexity of the standard repetition-based approach. This bottleneck disappears for regular OWFs, whose output distributions are already flat on their image, and helps explain the sharp quantitative gap between the regular and general cases.

From OWFs to UOWHFs. Universal one-way hash functions were introduced by Naor and Yung [38], who constructed them from one-way permutations. De Santis and Yung [22] extended the construction to regular one-way functions, and Rompel [39] subsequently showed that arbitrary one-way functions suffice, thereby establishing the equivalence between OWFs and UOWHFs at the level of polynomial security. Katz and Koo [34] later provided a complete treatment of Rompel’s construction, correcting several gaps and omissions in the original presentation. Haitner, Holenstein, Reingold, Vadhan, and Wee [27] gave a conceptually simpler construction from arbitrary OWFs, based on inaccessible entropy, with improved efficiency and security. A parallel line of work obtained substantially more efficient constructions from regular and almost-regular OWFs, culminating in nearly optimal and simple constructions [48, 36]. More recently, Mao, Mazon, and Zhang [35] gave the first non-adaptive construction from arbitrary OWFs. As in the PRG case, however, the known transformations from arbitrary OWFs incur polynomial overhead and therefore do not turn an exponentially hard OWF into a UOWHF that is exponentially secure as a function of its message length.

Highly efficient cryptographic constructions. A separate line of work studies how efficiently cryptographic primitives can be realized. In the sequential setting, Ishai, Kushilevitz, Ostrovsky, and Sahai (IKOS) [33] initiated the study of cryptography with constant computational overhead (i.e., computable by linear-size circuits), and in particular developed linear-size implementations of basic information-theoretic tools such as pairwise-independent hashing; this direction was further developed in subsequent works. In the parallel setting, Applebaum, Ishai, and Kushilevitz (AIK) showed that many cryptographic primitives can be compiled into NC^0 , using randomized encodings and related techniques [7, 6]. Overall, both AIK and IKOS achieve very strong forms of computational efficiency, but their generic transformations do not provide the exponential-security-preserving OWF-to-PRG/UOWHF reductions sought here.

Preserving exponential security. The question of simultaneously achieving very low complexity and exponential security was studied by Baron, Ishai, and Ostrovsky [15]. They showed that an exponentially hard, linear-size computable OWF yields an exponentially secure, linear-size PRG with linear stretch, provided that the OWF additionally satisfies an entropy or regularity condition. In fact, they provided a linear-size black-box transformation. Applebaum [4], using new families of randomized encodings, showed that this black-box transformation can also be made locally computable while preserving linear size. In particular, an exponentially hard *almost-regular* local OWF with linear output length yields an exponentially secure local PRG with linear stretch. In the non-local setting, efficient UOWHF constructions from regular and almost-regular OWFs are also known [48, 36], and preserve exponential security when instantiated with exponentially hard OWFs and appropriate parameters. The present work removes the regularity assumption in the local setting. In contrast to the preceding black-box transformations, our transformation is non-black-box: it exploits the local circuit structure of the underlying OWF to transform an arbitrary exponentially hard local OWF into one with sufficient regularity, after which the existing local black-box transformations can be applied.

Local cryptography from concrete assumptions. Another line of work constructs local cryptographic primitives from more concrete assumptions. For PRGs, it is shown in [2] that the one-wayness of Goldreich-

style random local functions implies local PRGs with long stretch, via a search-to-decision reduction. This approach has been sharpened and extended in subsequent works [14, 11, 19, 18, 41]. (See [3] for a survey of the broader literature on the cryptographic hardness of random local functions.) Local UOWHFs with linear shrinkage were studied by Applebaum and Moses [12, 13]; the correction [13] identifies a gap in the claimed reduction from random local one-wayness, while leaving the construction valid under a stronger target-collision-resistance assumption. Low-complexity collision-resistant hash functions (which are stronger than UOWHFs) were constructed from concrete coding- and algebraic-style assumptions by Applebaum, Haramaty-Krasne, Ishai, Kushilevitz, and Vaikuntanathan [5].

1.3 AI Disclosure

OpenAI’s ChatGPT (5.5 and 5.6) was used during the preparation of this manuscript as an interactive assistant for improving exposition and organization, discussing and pruning proof strategies, and locating or checking bibliographic information. All mathematical arguments, technical claims, and references were independently verified by the author, and all AI-assisted text was reviewed and edited by the author. The author takes full responsibility for the content of the paper.

2 Preliminaries

All logarithms are to base two. We write $[n] = \{1, \dots, n\}$, U_n for the uniform distribution on $\{0, 1\}^n$, and $\oplus$ for bitwise addition over $\mathbb{F}_2$. We identify bit strings with vectors over $\mathbb{F}_2$ when discussing linear maps. The Hamming distance between a pair of strings $x, x' \in \{0, 1\}^n$, denoted by $\text{Ham}(x, x')$ is the number of locations in which x differs from x' . An *input restriction* of a function $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$ is a partial assignment $R : T \rightarrow \{0, 1\}$, where $T = \text{dom}(R) \subseteq [n]$. For $x \in \{0, 1\}^n$, we write $R[x] \in \{0, 1\}^n$ for the string obtained by overwriting the coordinates in T according to R and leaving all other coordinates unchanged, and write $f_R(x) = f(R[x])$.

Distribution ensembles and indistinguishability. A *distribution ensemble* $Y = \{Y_n\}_{n \geq 1}$ is a sequence of distributions indexed by n , typically over strings of polynomially bounded length. We often suppress the index when it is clear from context. An ensemble is *efficiently samplable* if a probabilistic polynomial-time algorithm, on input 1^n , samples from Y_n . We do not assume efficient samplability unless explicitly stated.

Two ensembles $Y = \{Y_n\}_{n \geq 1}$ and $Z = \{Z_n\}_{n \geq 1}$, over strings of the same length at each n , are (T, ε) -*indistinguishable* if, for all sufficiently large n and every probabilistic Boolean circuit D of size at most $T(n)$,

$$|\Pr[D(Y_n) = 1] - \Pr[D(Z_n) = 1]| \leq \varepsilon(n).$$

Their statistical distance, denoted $\text{SD}(Y_n, Z_n)$, is the maximum of this distinguishing advantage over all tests, without a size bound. We use $\approx_c$ and $\approx_s$ for computational and statistical indistinguishability, respectively, with the parameters specified in context.

Functions and collections. When we write a function $f : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}$, or simply $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$, we implicitly refer to an infinite sequence $\{f_n\}_{n \geq 1}$, with one function $f_n : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}$ for every input length. We typically suppress both the index n and the dependence of parameters such as m on n . We say that f is *efficiently computable* if there is a deterministic polynomial-time algorithm that, given 1^n , outputs a circuit computing f_n .

Similarly, a *collection* is an infinite sequence of keyed families

$$\mathcal{F} = \{f_k : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}\}_{k \in \{0, 1\}^{t(n)}},$$

where the dependence on n is again implicit. The collection is *efficiently computable* if, given $(1^n, k)$, a circuit computing f_k can be generated in polynomial time. Equivalently, the collection evaluation function

$$F_n : \{0, 1\}^n \times \{0, 1\}^{t(n)} \rightarrow \{0, 1\}^{m(n)},$$

that maps (x, w) to $f_{n,w}(x)$ is efficiently computable. We usually suppress n and write $\{f_w\}_w$ or $f(x, w)$. Unless specified otherwise, the key is sampled uniformly from $\{0, 1\}^{t(n)}$.

Output locality, input locality, and essential support. For function $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$ write $f = (f_1, \dots, f_m)$. The *essential support* of output f_j is

$$E_j = \text{esssupp}(f_j) = \{i \in [n] : \exists x \in \{0, 1\}^n, f_j(x) \neq f_j(x \oplus e_i)\},$$

where e_i is the i th unit vector. Thus essential support records actual dependence, not merely the variables occurring in a circuit description. The *output locality* of f is $\max_j |E_j|$. The *input locality* of a function is

$$\max_{i \in [n]} |\{j \in [m] : i \in E_j\}|,$$

namely, the maximum number of outputs influenced by one input coordinate [9]. A function $f = \{f_n\}$ has constant output locality (resp., input locality) if for some constant d and every n the function f_n has output locality (resp., input locality) of at most d . The class of functions that have constant output locality is exactly the complexity class NC^0 . By default, *locality* means output locality, and a function is d -local if this quantity is at most d . Under an input restriction, supports and localities refer to the resulting restricted functions. For a keyed family, locality for each fixed key refers only to its message inputs; when locality is required jointly in the message and key, we say so explicitly.

2.1 Cryptographic primitives

Definition 2.1 (One-way functions). *An efficiently computable function $f : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}$ is (T, ε) -one-way if every probabilistic circuit A of size at most $T(n)$ satisfies*

$$\Pr_{X \leftarrow \mathcal{U}_n} [A(f(X)) \in f^{-1}(f(X))] \leq \varepsilon(n).$$

Note that the adversary is allowed to be non-uniform. Indeed, in contrast to the uniformity required of the constructions, security is always against *nonuniform* probabilistic Boolean circuits.

Definition 2.2 (Pseudorandom generators). *An efficiently computable function $G : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}$ is a (T, ε) -secure pseudorandom generator (PRG) if $m(n) > n$ and $G(\mathcal{U}_n)$ and $\mathcal{U}_{m(n)}$ are (T, ε) -indistinguishable. Its additive stretch is $m(n) - n$. We say it has linear stretch, or constant multiplicative stretch, if $m(n) - n = \Omega(n)$ and $m(n) = O(n)$.*

Definition 2.3 (Universal one-way hash functions). *An efficient collection*

$$\mathcal{H} = \{h_k : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}\}_{k \in \{0, 1\}^{t(n)}}, \quad m(n) < n,$$

is a (T, ε) -secure universal one-way hash function (UOWHF) if every two-stage probabilistic adversary (A_1, A_2) of total circuit size at most $T(n)$ succeeds with probability at most $\varepsilon(n)$ in the experiment

$$(x, \text{st}) \leftarrow A_1(1^n), \quad K \leftarrow \mathcal{U}_{t(n)}, \quad x' \leftarrow A_2(x, \text{st}, K),$$

where success means $x, x' \in \{0, 1\}^n$, $x' \neq x$, and $h_K(x') = h_K(x)$. In particular, the target x is chosen before the key is revealed. This is the target-collision-resistance notion of Naor and Yung [38]. The key is public and is not counted as part of the digest. The additive shrinkage is $n - m(n)$; it is linear when $n - m(n) = \Omega(n)$.

Exponential security. An OWF is *exponentially hard*, and a PRG or UOWHF is *exponentially secure* (also, *exponentially strong*), if it satisfies its respective definition with

$$T(n) = 2^{\beta n}, \quad \varepsilon(n) = 2^{-\beta n}$$

for some constant $\beta > 0$. Here n denotes the OWF input length, the entire PRG seed length, or the UOWHF message length, respectively. Thus all random bits included in the PRG seed, or in the message of an encoded UOWHF, count toward this length. Our final UOWHF constructions also have $O(n)$ -bit keys, so the message length and the combined message/key length differ by only a constant factor.

The following standard proposition shows that PRGs and UOWHFs directly give rise to one-way functions.

Proposition 2.4 (PRGs and UOWHFs imply OWFs). *The following implications hold.*

1. If $G : \{0, 1\}^n \rightarrow \{0, 1\}^m$ is a (T, ε) -secure PRG, then G is a $(T - S - O(m), \varepsilon + 2^{n-m})$ -one-way function, where S is the circuit complexity of G .
2. Let $\mathcal{H} = \{h_k : \{0, 1\}^n \rightarrow \{0, 1\}^m\}_{k \in \{0, 1\}^t}$ be a (T, ε) -secure UOWHF. Then

$$F(k, x) = (k, h_k(x))$$

is a $(T - S - O(n), \varepsilon + 2^{m-n})$ -one-way function, where S is the circuit complexity of the evaluation map $(k, x) \mapsto h_k(x)$.

Consequently, a bitwise-optimal PRG or a bitwise-optimal UOWHF immediately yields a bitwise-optimal OWF; in the UOWHF case, we use that $t = O(n)$ and that the evaluation map is jointly constant-local in the message and key bits.

Proof. For the first item, suppose that A inverts G with probability q . Given a challenge $z \in \{0, 1\}^m$, distinguish by running $A(z)$ and accepting iff $G(A(z)) = z$. On input $G(U_n)$ this test accepts with probability q , whereas on U_m it accepts only if the challenge lies in $\text{Im}(G)$, which happens with probability at most 2^{n-m} . Hence $q \leq \varepsilon + 2^{n-m}$.

For the second item, suppose that A inverts F with probability q . Construct a UOWHF adversary as follows. In its first stage it samples $X \leftarrow U_n$ and chooses X as its target. After receiving the public key K , it computes $Y = h_K(X)$, runs $A(K, Y)$, and returns the x -component X' of its answer. Whenever A successfully inverts F and $X' \neq X$, this gives a target collision. It remains to bound the probability of successful inversion with $X' = X$.

Fix a key k and the randomness r of A , and let $\phi_{k,r}(y)$ denote the x -component returned by $A(k, y)$. For each $y \in \text{Im}(h_k)$, there is at most one

$$x \in h_k^{-1}(y)$$

for which $\phi_{k,r}(y) = x$. Consequently,

$$|\{x \in \{0, 1\}^n : \phi_{k,r}(h_k(x)) = x\}| \leq |\text{Im}(h_k)| \leq 2^m.$$

Thus, for uniform X ,

$$\Pr[\phi_{k,r}(h_k(X)) = X] \leq 2^{m-n}.$$

Averaging over K and the randomness of A gives the same bound. Therefore the UOWHF adversary finds a target collision with probability at least

$$q - 2^{m-n},$$

and hence $q \leq \varepsilon + 2^{m-n}$. □

2.2 Fibers and regularity

For a function $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$ and $y \in \text{Im}(f)$, define the *fiber size*, *log-fiber size* and *sample entropy* of y by

$$K_f(y) := |f^{-1}(y)|, \quad r_f(y) := \log_2 K_f(y), \quad \iota_f(y) := -\log_2 \Pr[f(U_n) = y].$$

Observe that $\iota_f(y) = n - r_f(y)$ and $\Pr[f(U_n) = y] = 2^{-\iota_f(y)}$. We extend these quantities to inputs $x \in \{0, 1\}^n$ by writing

$$K_f(x) := K_f(f(x)), \quad r_f(x) := r_f(f(x)), \quad \iota_f(x) := \iota_f(f(x)).$$

Thus $K_f(x)$ is the size of the fiber containing x , while $\iota_f(x)$ is the sample entropy of its output $f(x)$. All three quantities are constant on fibers. A *regularity layer* is a set

$$L_j(f) = \{x \in \{0, 1\}^n : j \leq r_f(f(x)) < j + 1\}, \quad j \in \{0, \dots, n\}.$$

It is a union of complete fibers, and the sizes of its fibers differ by a factor of less than two.

Definition 2.5 (Typical almost-regularity). *For $\alpha \geq 0$ and $\delta = \delta(n) \in [0, 1]$, a function $f : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}$ is (α, δ) -typically almost regular if there is a number $\ell = \ell(n) \in [0, n]$ such that*

$$\Pr_{X \leftarrow U_n} [\ell \leq r_f(X) \leq \ell + \alpha n] \geq 1 - \delta(n).$$

The interval $[\ell, \ell + \alpha n]$ is a regularity window, and its lower endpoint is the regularity advice. We say that f is typically α -almost regular when the exceptional mass is at most 2^{-cn} for some constant $c > 0$. When $\delta = 0$, this is the usual α -almost-regularity condition used in [4]. A typically almost-regular OWF is an OWF having this additional property; one-wayness is still required for an unconditioned uniform input.

Whenever integer regularity advice is convenient, we may round ℓ down; this enlarges the regularity window by less than one, which is negligible compared to the linear window widths considered here. Moreover, membership in a regularity window depends only on the output $f(x)$. Hence the corresponding good set, $\{x : r_f(x) \in [\ell, \ell + \alpha n]\}$, is a union of complete fibers of f ; in particular, conditioning on this set does not alter the conditional distribution within any surviving fiber.

2.3 Randomized encoding

A randomized encoding of $f(x)$ is a randomized map $\hat{f}(x; r)$ from which $f(x)$ can be recovered, while revealing no additional information about x . We use perfect randomized encodings with the balanced-simulation and length-preservation requirements of [7] (see also [4, Section 2.3]).

Definition 2.6 (Perfect randomized encoding). *Let $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$. A map*

$$\hat{f} : \{0, 1\}^n \times \{0, 1\}^\rho \rightarrow \{0, 1\}^{\hat{m}}$$

is a perfect randomized encoding (PRE) of f if there are an efficient deterministic decoder Dec and an efficient randomized simulator Sim with the following properties.

1. Perfect correctness. *For every $x \in \{0, 1\}^n$ and $r \in \{0, 1\}^\rho$,*

$$\text{Dec}(\hat{f}(x; r)) = f(x).$$

2. Perfect privacy. *For every $x \in \{0, 1\}^n$,*

$$\hat{f}(x; U_\rho) \equiv \text{Sim}(f(x)).$$

3. Balanced simulation.

$$\text{Sim}(\mathbf{U}_m) \equiv \mathbf{U}_{\widehat{m}}.$$

4. Length preservation.

$$\widehat{m} - (n + \rho) = m - n, \quad \text{equivalently,} \quad \widehat{m} = m + \rho.$$

Here $\equiv$ denotes equality of distributions, including the simulator's internal randomness. We call r the random input and ρ the complexity, or randomness complexity, of the encoding.

The above requirements also imply the *unique-randomness property*: for fixed x , the map $r \mapsto \widehat{f}(x; r)$ is injective [7, Section 4]. We use this consequence when transferring target-collision resistance to an encoding.

Encoding families. The definition extends pointwise to a family $\{f_w : \{0, 1\}^n \rightarrow \{0, 1\}^m\}_{w \in \{0, 1\}^t}$: for every w , the map $\widehat{f}_w$ must perfectly encode f_w . We require a deterministic polynomial-time algorithm that, given $(1^n, w)$, outputs descriptions of $\widehat{f}_w$, its decoder, and its simulator. Encoding each fixed-key function should be distinguished from encoding the joint map $(x, w) \mapsto (f_w(x), w)$ with locality in all its inputs.

2.4 Pairwise Independent Hashing and List-Decodable Codes

Let $h : \mathcal{X} \times \mathcal{W} \rightarrow \mathcal{Z}$ be a two-argument function. We regard its first argument as the main input and its second argument as a *key*, or *index*. Write $h_w(x) = h(x, w)$ and view h as the family $\{h_w : \mathcal{X} \rightarrow \mathcal{Z}\}_{w \in \mathcal{W}}$. We call h *pairwise independent* if, for every distinct $x, x' \in \mathcal{X}$, the pair $(h_W(x), h_W(x'))$ is uniform on $\mathcal{Z}^2$ when W is uniform on $\mathcal{W}$.

As in [4, Section 2.4], we also regard h as a code sending x to the codeword $(h_w(x))_{w \in \mathcal{W}}$ of length $|\mathcal{W}|$ over alphabet $\mathcal{Z}$. Evaluation of h gives direct access to any codeword coordinate. A possibly randomized oracle $\mathcal{O} : \mathcal{W} \rightarrow \mathcal{Z} \cup \{\perp\}$ is (δ, ε) -*correlated* with the codeword of x if

$$\Pr[\mathcal{O}(w) \neq \perp] \geq \delta \quad \text{for every } w \in \mathcal{W},$$

and

$$\Pr_{W \leftarrow \mathcal{W}}[\mathcal{O}(W) = h_W(x) \mid \mathcal{O}(W) \neq \perp] \geq \frac{1}{|\mathcal{Z}|} + \varepsilon.$$

The probabilities include the oracle's randomness; W is uniform. We say that h is (δ, ε) -*list-decodable* with κ oracle calls and list size λ if an oracle algorithm, given access to any oracle correlated with x as above, makes at most κ calls and outputs a list $\Lambda \subseteq \mathcal{X}$ of size at most λ satisfying

$$\Pr[x \in \Lambda] \geq \frac{1}{2}.$$

Its additional running time is $\lambda \cdot \text{poly}(\log |\mathcal{X}|)$, not counting the cost of answering oracle queries.

Conditional key-samplability. We say that h satisfies the *conditional key-samplability* property, or *key-samplability* for short, if there is a probabilistic polynomial-time algorithm which, given distinct $x, x' \in \mathcal{X}$, samples exactly from

$$W \mid h_W(x) = h_W(x'), \quad W \leftarrow \mathcal{W}.$$

In other words, it samples a uniformly random key subject to a prescribed collision. The running time is polynomial in the input, key, and output representation lengths. For pairwise-independent hashing, this conditioning event has probability $1/|\mathcal{Z}|$, so it is always well defined; nevertheless, rejection sampling need not be efficient. The property requires uniformity over *all* consistent keys, not just finding any key that causes the collision.

Key-linearity. A Boolean hash family $h : \{0, 1\}^n \times \{0, 1\}^t \rightarrow \{0, 1\}^s$ is *key-linear* if, for every fixed x , the map $w \mapsto h(x, w)$ is linear over $\mathbb{F}_2$. Equivalently, for every x there is a matrix $L_x \in \mathbb{F}_2^{s \times t}$ such that

$$h(x, w) = L_x w.$$

This concerns linearity in the key, not in the main input.

The next theorem collects the hashing guarantees of [4, Theorem 3.2], the two-level encoding guarantee underlying the localization argument in the proof of [4, Corollary 5.3], and the key-linearity property of the affine IKOS/BIO implementation. We record only the proof of this additional property and its key-samplability consequence.

Theorem 2.7 (Locally encodable, key-samplable hashing). *For every output length $s = s(n) = O(n)$, there are $t = t(n) = O(n)$ and an efficiently computable function*

$$h : \{0, 1\}^n \times \{0, 1\}^t \rightarrow \{0, 1\}^s$$

with the following properties.

1. *h is pairwise independent.*
2. *For every $\delta, \varepsilon > 0$, its induced code is (δ, ε) -list-decodable with $O(n/(\delta\varepsilon^2))$ oracle calls and list size.*
3. *For every fixed key w , the function h_w has a three-local perfect randomized encoding with $O(n)$ randomness complexity. Moreover, the joint map*

$$(x, w) \mapsto (h(x, w), w)$$

also has such an encoding, with locality in both x and w and in all encoding coins. This encoding can be chosen to copy w verbatim to its output, with a randomness length independent of the value of w .

4. *Furthermore, the same encoding guarantee holds for the following two-level composition of instances of this hashing construction. Let H and E denote the instances of h with input/output lengths (n, s_1) and $(m + s_1, s_2)$, respectively, where $m, s_1, s_2 = O(n)$, using fixed input padding when necessary. Then the joint evaluation map*

$$(x, z, w_1, w_2) \mapsto (w_1, w_2, E_{w_2}(z, H_{w_1}(x)))$$

admits a three-local perfect randomized encoding with $O(n)$ randomness complexity. The encoding copies both keys verbatim to its output, and its randomness length is independent of their values. Locality holds jointly in x, z, w_1, w_2 and all encoding coins.

5. *h is key-linear and satisfies conditional key-samplability.*

The encodings of items 3 and 4 and their decoders and simulators are polynomial-time constructible. All $O(n)$ bounds above are uniform when the relevant length parameters are bounded by fixed constant multiples of n .

The hashing and encoding guarantees follow from the cited results and localization argument. We prove below the key-linearity assertion and its conditional key-samplability consequence.

Proof of the additional assertion 5. Use the affine realization of the IKOS construction described in [15]; see [33, Section 3.2] for the underlying construction. It applies a fixed input encoding with constant-size blocks $c_1(x), \dots, c_q(x)$, independently hashes the blocks by maps $z \mapsto A_i z + b_i$, and applies a fixed binary linear map L to the concatenated results. Thus its key can be taken to be the entries of $w = ((A_i, b_i))_{i \in [q]}$, and

$$h(x, w) = L(A_1 c_1(x) + b_1, \dots, A_q c_q(x) + b_q).$$

For fixed x , every $c_i(x)$ is constant, so this expression is linear in w . Fixed input padding and output projections preserve this property.

For completeness, we show that key-linearity yields an explicit conditional sampler. Given distinct x, x' , recover L_x and $L_{x'}$ by evaluating h at the standard basis keys: column j of L_x is $h(x, e_j)$. The keys causing the prescribed collision form exactly the linear subspace

$$\mathcal{K}_{x, x'} = \ker(L_x + L_{x'}).$$

Gaussian elimination computes a basis $v_1, \dots, v_r$ of this subspace in polynomial time. Output

$$W = \sum_{i=1}^r B_i v_i, \quad B_1, \dots, B_r \leftarrow U_1 \text{ independently.}$$

Each vector of $\mathcal{K}_{x, x'}$ has a unique representation in this basis, so W is uniform on $\mathcal{K}_{x, x'}$, exactly as required. Pairwise independence also implies that $L_x + L_{x'}$ has rank s , and hence $r = t - s$, since $h_W(x) + h_W(x')$ is uniform on $\{0, 1\}^s$.

The same argument gives the stronger ability to sample a uniform key conditioned on two prescribed values $h_W(x) = z$ and $h_W(x') = z'$: solve the corresponding affine linear system and sample a uniform translate of its kernel. For $x \neq x'$, pairwise independence ensures that the system is consistent for every $z, z' \in \{0, 1\}^s$. $\square$

3 Sparsification theorem

The following structural theorem shows one can take any local function f and keep only $O(n)$ of its outputs in a way that almost preserves the information. That is, after fixing a small fraction of the inputs, the entire restricted map factors through a linear-size projection of its original outputs via a constant-local decoder.

Theorem 3.1 (Sparsification theorem). *For every integer $d \geq 1$ and constant $0 < \rho < 1$, there is a constant $C_{d, \rho}$ with the following property. Every d -local map*

$$f : \{0, 1\}^n \longrightarrow \{0, 1\}^m$$

admits a restriction R fixing at most ρn input coordinates and a set $S \subseteq [m]$ of at most $C_{d, \rho} n$ original output coordinates such that, writing

$$g(x) = f(R[x])_S,$$

there is a map

$$D : \{0, 1\}^{|S|} \longrightarrow \{0, 1\}^m$$

for which

$$f(R[x]) = D(g(x)) \quad \text{for every } x \in \{0, 1\}^n. \quad (1)$$

Moreover, every output coordinate of D depends on at most $Q_d = 2^{2^d}$ coordinates of its input. In particular, for all x, z consistent with R ,

$$f(x)_S = f(z)_S \iff f(x) = f(z). \quad (2)$$

For fixed d, ρ , the restriction R , the set S , and the constant-local decoder D can be found in deterministic polynomial time from the NC^0 circuit of f .

We keep all n input coordinates when applying a restriction: for $x \in \{0, 1\}^n$, the string $R[x] \in \{0, 1\}^n$ is obtained by overwriting the coordinates fixed by R , and we write $f_R(x) = f(R[x])$. Thus the fixed coordinates remain formally present but are ignored by f_R .

Remark 3.2 (Why an input restriction is necessary: a quantitative tradeoff). *The restriction in Theorem 3.1 cannot in general be removed, even if the decoder is allowed to be arbitrary. More quantitatively, for every prescribed constant $C > 0$ there is a 2-local map for which any factorization through at most Cn original output coordinates requires fixing $\Omega_C(n)$ input coordinates. This can be shown by slightly tweaking the pairwise-product example from the introduction.*

Indeed, fix an integer $r \geq 2$ and partition the inputs into $N = n/r$ disjoint blocks of size r (assume that $n = rN$). In every block output all pairwise ANDs $x_i x_j$. Thus the map has

$$N \binom{r}{2} = \frac{r-1}{2} n$$

outputs. Let R fix t input coordinates, let S be a set of retained original outputs, and suppose that the full restricted output factors through the retained projection. At least $N - t$ blocks are untouched by R . In each untouched block every pairwise-AND coordinate must be retained: if $x_i x_j$ were omitted, the all-zero assignment and the assignment having exactly $x_i = x_j = 1$ would agree on every retained coordinate but disagree on this omitted output. Hence

$$|S| \geq (N - t) \binom{r}{2}.$$

If $|S| \leq Cn = CrN$, then

$$t \geq N \left(1 - \frac{2C}{r-1} \right). \quad (3)$$

Taking, for example, an integer $r \geq 4C + 1$ gives $t \geq N/2 = \Omega_C(n)$. Equivalently, suppose the allowed restriction has size at most ρn . For sufficiently small ρ , choose an integer r with $1/(4\rho) \leq r \leq 1/(2\rho)$. Then at least half the blocks are untouched, so every such factorization has

$$|S| \geq \frac{r-1}{4} n = \Omega(n/\rho).$$

Thus the constant $C_{d,\rho}$ must grow at least as $\Omega(1/\rho)$ as $\rho \rightarrow 0$; its dependence on ρ is inherent, not merely an artifact of the proof.

The proof of the sparsification theorem will be given in Section 3.2. The proof employs a simple “agreement” lemma that is stated and proved in Section 3.1.

A simple adaptation of the proof of the sparsification theorem yields the following corollary whose proof is deferred to Section A.

Corollary 3.3 (Entropy-preserving compression of local sources). *For every integer $d \geq 1$ and constant $0 < \rho < 1$, every d -local map $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$ admits an input restriction R fixing at most ρn coordinates and a set $S \subseteq [m]$ of $O_{d,\rho}(n)$ output coordinates such that*

$$f(R[x]) = D(f(R[x])_S) \quad \text{for every } x \in \{0, 1\}^n,$$

where $D : \{0, 1\}^{|S|} \rightarrow \{0, 1\}^m$ has output locality at most $Q_d = 2^{2^d}$. Moreover, writing $Y = f(U_n)$ and $Y_R = f(R[U_n])$, we have

$$H((Y_R)_S) = H(Y_R) \geq H(Y) - \rho n.$$

Thus, at the cost of fixing at most ρn bits of the underlying randomness and losing at most ρn bits of Shannon entropy, the source admits exact projection-based compression with constant-local reconstruction.

3.1 An agreement lemma

The following lemma shows that, in order to prove that a function $h : \{0, 1\}^E \rightarrow \{0, 1\}$ takes the same value on two inputs $x, z \in \{0, 1\}^E$, it suffices to establish agreement under a suitable collection of restrictions of h , provided that these restrictions have a simple hierarchical structure. Intuitively, each restriction specifies a block of coordinates that may be switched together, and the combinatorial condition below ensures that these blocks form a transitive dependency relation, allowing them to be eliminated one at a time.

Lemma 3.4 (Hierarchical agreement). *Let E be finite, let $h : \{0, 1\}^E \rightarrow \{0, 1\}$, and fix $x, z \in \{0, 1\}^E$. Suppose a set $A_i \subseteq E$ is assigned to every $i \in E$, with the following preorder condition:*

$$i \in A_i, \quad j \in A_i \implies A_j \subseteq A_i. \quad (4)$$

Suppose also that each assigned set can be switched as a whole:

$$h(x_{A_i}, b) = h(z_{A_i}, b) \quad \text{for every } b \in \{0, 1\}^{E \setminus A_i}. \quad (5)$$

Then $h(x) = h(z)$.

Intuition. The preorder condition (4) has a simple interpretation. Define $j \preceq i$ whenever $j \in A_i$. Then $\preceq$ is a preorder and A_i is the principal down-set of i . Thus, after contracting equivalent coordinates, the sets A_i form the dependency closures of a DAG. The lemma says that if every node can be switched together with all its dependencies, independently of the remaining coordinates, then the whole assignment can be switched. The proof simply eliminates a minimal dependency block at a time.

Proof. Induct on $|E|$. The empty case is immediate. Choose an inclusion-minimal set B among the A_i 's. For every $j \in B$, (4) and minimality give $A_j = B$. Every A_i is therefore either disjoint from B or contains all of B : if $j \in A_i \cap B$, then $B = A_j \subseteq A_i$.

Fix the coordinates in B to x_B , obtaining a function h_B on $E \setminus B$. For each $i \notin B$ let $A'_i = A_i \setminus B$. These nonempty sets still satisfy (4). Their switching identities for h_B follow directly from (5) when A_i is disjoint from B . When $A_i \supseteq B$, they follow by switching A_i and then switching B back:

$$h(x_B, x_{A_i \setminus B}, b) = h(z_B, z_{A_i \setminus B}, b) = h(x_B, z_{A_i \setminus B}, b).$$

Induction gives $h_B(x_{E \setminus B}) = h_B(z_{E \setminus B})$. Switching B once more identifies the latter value with $h(z)$, whereas the former is $h(x)$. $\square$

3.2 Proof of Thm. 3.1

3.2.1 Overview

The case $d = 1$ follows by retaining one copy of each nonconstant unary output, so assume $d \geq 2$. Write $E_j = \text{esssupp}(f_j)$, and put $Q_d = 2^{2^d}$ and $L_d = 2^d Q_d$. The proof proceeds in two steps. First, we sparsify f to a function G with $O_{d,\rho}(n)$ outputs from which the entire value $f(x)$ can be locally reconstructed for every input x . At this stage G is not necessarily a projection of f : in addition to original output functions f_j , its coordinates may include auxiliary Boolean functions obtained by hardwiring some of the local inputs of an individual output to fixed constants. For example, if $f_j(x) = \text{AND}(x_1, x_2, x_3)$, we may use the auxiliary function $\text{AND}(x_1, x_2, 1)$; this does not restrict the actual input, and x_3 may still take an arbitrary value. Thus G is defined on the entire input cube, and the first step yields a factorization

$$f = \Phi \circ G$$

that holds for every input.

To construct G , we build a family $\mathcal{A}$ of $O_{d,\rho}(n)$ candidate sets of input coordinates, each of size at most d . We include in G all nonconstant original output functions whose essential support is contained in some $A \in \mathcal{A}$, together with selected auxiliary functions of the above form, also supported on members of $\mathcal{A}$. Since there are at most Q_d Boolean functions supported on any one candidate, removing duplicates leaves at most $Q_d|\mathcal{A}| = O_{d,\rho}(n)$ outputs. The main combinatorial argument shows that these functions suffice for reconstruction and, moreover, that Φ can be taken to be Q_d -local.

In the second step, we eliminate the non-projective coordinates of G . We choose a random input restriction and extend it to repair the remaining exceptions, obtaining a restriction R that fixes at most ρn coordinates and makes each additional output of G_R either constant or equal to a restricted original output. For each $A \in \mathcal{A}$, we then retain one representative of every distinct nonconstant restricted original output whose essential support is contained in A , and remove duplicates globally. Let S be the resulting set of original output coordinates. Every coordinate of G_R is now either constant or represented in $(f_R)_S$, so the identity $f_R = \Phi \circ G_R$ yields $f_R = D \circ (f_R)_S$ for a Q_d -local decoder D , with $|S| \leq Q_d|\mathcal{A}| = O_{d,\rho}(n)$. Importantly, the candidate family $\mathcal{A}$ is constructed before, and independently of, the choice of R . Finally, we derandomize the choice of the restriction by conditional expectations.

3.2.2 Building the collection $\mathcal{A}$

Sections and petal covers. For $\emptyset \neq B \subseteq E_j$ and $b \in \{0, 1\}^{E_j \setminus B}$, the function

$$q : \{0, 1\}^B \longrightarrow \{0, 1\}, \quad q(u) = f_j(u, b),$$

is a *section* of f_j on B . We identify equal pairs (B, q) , allowing q to have inessential variables in its declared domain B . When comparing functions, we view them as functions on the full input cube, ignoring coordinates outside their domains. The sets $E_j \setminus B$, over all original outputs admitting the section (B, q) , are its *petals*.

For every section that is neither constant nor equal to an original output, choose a maximal collection of pairwise disjoint petals, and let $C_{B,q}$ be their union. We call $C_{B,q}$ the *petal cover* of (B, q) . All these petals are nonempty: an empty petal would make q an original output. No petal cover is needed for constants or original output functions. Fix a threshold $K \geq 1$, to be chosen later depending only on d and ρ . A section with a petal cover is *light* if $|C_{B,q}| < K$ and *heavy* otherwise.

Claim 3.5 (Petal-cover property). *If $C_{B,q}$ is defined, then every original output f_j admitting q on B satisfies*

$$C_{B,q} \cap (E_j \setminus B) \neq \emptyset.$$

Moreover, if the chosen collection contains k petals, then $|C_{B,q}| \leq (d-1)k$.

Proof. A petal disjoint from $C_{B,q}$ could be added to the chosen collection, contradicting maximality. Each chosen petal has size at most $d-1$. $\square$

The candidate family. Start with all singleton sets $\{1\}, \dots, \{n\}$. Whenever $A \in \mathcal{A}$ has $|A| < d$ and (B, q) is a light section with $\emptyset \neq B \subseteq A$, add

$$A \cup \{v\} \quad \text{for every } v \in C_{B,q} \setminus A.$$

Continue until no new candidates are produced. This construction depends only on f , the chosen petal covers, and K ; no restriction has yet been chosen.

Claim 3.6 (Number of candidates). *The candidate family satisfies*

$$|\mathcal{A}| \leq B_K n, \quad B_K := \sum_{t=0}^{d-1} (L_d K)^t.$$

Proof. For each candidate there are at most 2^d possible nonempty subsets B and at most Q_d functions on each such subset. Thus at most L_d section pairs can trigger extensions, each contributing fewer than K new coordinates. Every growth path starts at a singleton and has length at most $d - 1$, giving the stated bound. $\square$

3.2.3 The non-projective sparsification G

We are now ready to define the intermediate mapping G . Let

$$\mathcal{T} := \{(B, q) : (B, q) \text{ is heavy and } B \subseteq A \text{ for some } A \in \mathcal{A}\}.$$

There are at most L_d relevant section pairs per candidate, so

$$|\mathcal{T}| \leq L_d |\mathcal{A}| \leq L_d B_K n. \quad (6)$$

Temporarily allow these heavy sections as additional output functions. Specifically, let G list, without repetition, all nonconstant original output functions supported on a member of $\mathcal{A}$, together with all the functions q indexed by $\mathcal{T}$. Every coordinate of G is supported on a candidate, so G has at most $Q_d |\mathcal{A}|$ coordinates. Although G need not be a projection of f , it already suffices for reconstruction.

Lemma 3.7 (Reconstruction from candidate outputs and heavy sections). *Every original output f_j is determined by the coordinates of G whose essential support is contained in E_j . Consequently, there is a Q_d -local map Φ such that*

$$f = \Phi \circ G. \quad (7)$$

Proof. Fix $h = f_j$ and $E = E_j$; the assertion is immediate if h is constant. For each $i \in E$, construct $A_i \subseteq E$ by starting with $A_i = \{i\}$ and repeatedly applying any available candidate-growth step that adds a coordinate from E , until no such step remains. Since $|E| \leq d$, every intermediate set belongs to $\mathcal{A}$. We will need the following claims.

Claim 3.8 (Preorder property). *The sets $\{A_i\}_{i \in E}$ satisfy the preorder condition of Lemma 3.4; namely,*

$$i \in A_i, \quad r \in A_i \implies A_r \subseteq A_i.$$

Proof. The first property is immediate from the definition of A_i . For the second, fix $r \in A_i$ and consider the growth process that constructs A_r from the singleton $\{r\}$. We claim inductively that every set produced in this process is contained in A_i . This is true initially since $r \in A_i$.

Suppose the current set $A' \subseteq A_i$, and the next growth step adds $v \in C_{B,q} \cap E$ for some light section (B, q) with $\emptyset \neq B \subseteq A'$. Then $B \subseteq A_i$. If $v \notin A_i$, the same growth step would be available from A_i , contradicting the termination of its construction. Hence $v \in A_i$. Thus every coordinate added in the construction of A_r also belongs to A_i , and therefore $A_r \subseteq A_i$. $\square$

Claim 3.9 (The sections are available). *Every section of h on A_i is either constant, an original output function, or heavy. Consequently, each of these sections is either constant or appears as a coordinate of G supported on A_i .*

Proof. Indeed, otherwise, such a section q would be light, and Claim 3.5 would give

$$C_{A_i, q} \cap (E \setminus A_i) \neq \emptyset.$$

But then another candidate-growth step within E would be possible, contradicting the termination of the construction of A_i . Hence every nonconstant section of h on A_i appears as a coordinate of G supported on A_i : original output functions are included by the definition of G , and heavy sections are included because $A_i \in \mathcal{A}$. $\square$

Now suppose $x, z \in \{0, 1\}^n$ satisfy $g(x) = g(z)$ for every coordinate function g of G supported on E . For every $i \in E$ and every $b \in \{0, 1\}^{E \setminus A_i}$, the section $u \mapsto h(u, b)$ is either constant or one of these coordinates. Thus

$$h(x_{A_i}, b) = h(z_{A_i}, b).$$

The hierarchical agreement lemma (Lemma 3.4) implies $h(x_E) = h(z_E)$, proving the claimed determination property.

Letting

$$T_j := \{s \in [|G|] : \text{esssupp}(G_s) \subseteq E_j\}$$

consist of those coordinates of G whose corresponding Boolean functions depend only on variables from the support E_j of f_j , we have shown that

$$G(x)_{T_j} = G(z)_{T_j} \implies f_j(x) = f_j(z).$$

Hence there is a Boolean function

$$\Phi_j : \{0, 1\}^{T_j} \rightarrow \{0, 1\} \quad \text{such that} \quad f_j(x) = \Phi_j(G(x)_{T_j})$$

for every x . Since $|E_j| \leq d$, there are at most $2^{|E_j|} \leq Q_d$ distinct Boolean functions whose essential support is contained in E_j . As G contains no duplicate functions, $|T_j| \leq Q_d$. Combining the maps Φ_j over all output coordinates gives a Q_d -local map Φ such that $f = \Phi \circ G$. The lemma follows. $\square$

3.2.4 Deriving a projection-based sparsification for a random restriction

The factorization (7) remains valid under every restriction R :

$$f_R = \Phi \circ G_R.$$

It therefore remains to choose a small restriction under which every heavy section in $\mathcal{T}$ becomes either constant or equal to a restricted original output. All coordinates of G_R will then be available from the retained projection.

Choosing the restriction. Fix a rational $0 < p \leq \rho/4$. In a random restriction R_0 , each coordinate is independently left free with probability $1 - p$ and fixed to each bit with probability $p/2$. For any section q on B , write q_{R_0} for the function obtained by applying R to the coordinates of B .

For each $(B, q) \in \mathcal{T}$ and each of its chosen petals, choose a witnessing original output f_s and an assignment to $E_s \setminus B$ that produces q . If R_0 fixes this entire petal to its witnessing assignment, then $q_{R_0} = (f_s)_{R_0}$. This equality persists under every further restriction. Let $\text{Fail}_{B, q}$ be the event that none of the chosen petals is fixed to its witnessing assignment. We will show that this event happens with probability that is exponentially small in the parameter K .

Let $P_1, \dots, P_k$ be the chosen pairwise-disjoint petals whose union is $C_{B,q}$. For each t , the probability that R_0 fixes the entire petal P_t to its designated witnessing assignment is

$$\left(\frac{p}{2}\right)^{|P_t|} \geq \left(\frac{p}{2}\right)^{d-1}.$$

Since the petals are disjoint, these events are independent. Hence

$$\Pr[\text{Fail}_{B,q}] = \prod_{t=1}^k \left(1 - \left(\frac{p}{2}\right)^{|P_t|}\right) \leq \left(1 - \left(\frac{p}{2}\right)^{d-1}\right)^k.$$

Moreover, since every chosen petal has size at most $d-1$,

$$|C_{B,q}| = \sum_{t=1}^k |P_t| \leq (d-1)k,$$

and therefore $k \geq |C_{B,q}|/(d-1)$. Using $1 - a \leq e^{-a}$, we obtain

$$\Pr[\text{Fail}_{B,q}] \leq \exp\left(-\frac{(p/2)^{d-1}}{d-1}|C_{B,q}|\right) = e^{-\kappa|C_{B,q}|} \leq e^{-\kappa K}, \quad \kappa := \frac{(p/2)^{d-1}}{d-1}, \quad (8)$$

where the last inequality uses that (B, q) is heavy.

Now choose the constant $K = K(d, \rho)$ large enough that

$$dL_d B_K e^{-\kappa K} \leq \rho/4. \quad (9)$$

Such a choice exists because B_K grows polynomially in K , whereas $e^{-\kappa K}$ decays exponentially.

Repairing failures. After sampling R_0 , for every $(B, q) \in \mathcal{T}$ for which $\text{Fail}_{B,q}$ occurs, fix every still-free coordinate of B to zero. Let R be the resulting restriction. Every section indexed by $\mathcal{T}$ is now either constant or equal to a restricted original output: failed sections become constant, and the witnessing equalities for successful sections remain valid. The number of fixed coordinates is at most

$$Z(R_0) := |\text{dom } R_0| + d \sum_{(B,q) \in \mathcal{T}} \mathbf{1}\{\text{Fail}_{B,q}\}. \quad (10)$$

By (6)–(9),

$$\mathbb{E}Z \leq pn + dL_d B_K n e^{-\kappa K} \leq \rho n/2.$$

Thus some choice of R_0 yields a repaired restriction fixing at most ρn coordinates. We give a deterministic choice below.

Retaining original outputs. For each $A \in \mathcal{A}$, retain one representative of every distinct nonconstant restricted original output whose essential support is contained in A , and remove duplicates globally. Let $S \subseteq [m]$ be the resulting set of output coordinates. Then

$$|S| \leq Q_d |\mathcal{A}| \leq Q_d B_K n.$$

Every coordinate of G_R is either constant or is equal, as a Boolean function, to one of the retained restricted original outputs $(f_s)_R$, $s \in S$. For a coordinate of G that is already an original output, this follows directly from the retention rule, since restriction can only shrink its essential support. For a heavy section $(B, q) \in \mathcal{T}$,

the construction of R makes q_R either constant or equal to a restricted original output whose essential support is contained in $B \subseteq A$ for some $A \in \mathcal{A}$; hence, if q_R is nonconstant, the retention rule keeps an output in S computing exactly the same function.

Substituting these constants and representatives into $f_R = \Phi \circ G_R$ gives

$$f_R = D \circ (f_R)_S.$$

Each output of Φ depends on at most Q_d coordinates, and each such coordinate is replaced by a constant or a single retained coordinate. Thus D is still Q_d -local. Taking $C_{d,\rho} = Q_d B_K$ proves the existence and decomposition assertions of the theorem.

3.2.5 Derandomizing the restriction

The conditional expectation of Z in (10) is computable exactly after any partial choice of the restriction symbols in $\{*, 0, 1\}$. For each $(B, q) \in \mathcal{T}$, its conditional failure probability is a product over its disjoint chosen petals. Each factor is one minus the conditional probability that the petal is fixed to its witnessing assignment. This latter probability is zero if an already chosen symbol is $*$ or a bit inconsistent with the assignment; otherwise it is $(p/2)^v$, where v is the number of undecided symbols in the petal. Independence between different section pairs is unnecessary, since their contributions to Z are summed.

Choose the symbols of R_0 one at a time, always taking a value that does not increase the conditional expectation of Z . The completed choice satisfies $Z(R_0) \leq \mathbb{E}Z \leq \rho n/2$, and hence its repair respects the restriction budget. All calculations have polynomial bit complexity for rational p . The cutoff can also be selected using rational arithmetic: with $a_0 = (p/2)^{d-1}$, it suffices to test

$$dL_d B_K (1 - a_0/2)^{\lfloor K/(d-1) \rfloor} \leq \rho/4,$$

which implies (9) because $e^{-a_0} \leq 1 - a_0/2$ for $0 \leq a_0 \leq 1$. For an arbitrary real budget ρ , first work with a smaller positive rational budget.

All sections, petals, petal covers, candidate sets, and retained functions are computable in deterministic polynomial time from the NC^0 circuit computing f . Once R and S are fixed, for each original output f_j enumerate the at most 2^d assignments to E_j consistent with R . This determines $(f_j)_R$ as a function of the retained restricted outputs whose essential supports are contained in E_j ; the reconstruction property proved above guarantees that this function is well defined. On tuples of retained-output values that never arise, define the function arbitrarily. This constructs the local decoder D_j , and hence D . This completes the deterministic construction and the proof of the theorem. $\square$

4 Deriving typical almost-regularity

In this section we prove that every exponentially-strong local OWF can be transformed into a typically almost-regular, exponentially strong local OWF. (See Section 2.2 for the regularity-related notation.)

Theorem 4.1 (Typically almost-regular local one-way functions). *Fix an integer $d \geq 1$ and constants $0 < \rho < \beta \leq 1$. Let $f : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}$, with $m(n) = \text{poly}(n)$, be a d -local, $(2^{\beta n}, 2^{-\beta n})$ -one-way function. There is a deterministic polynomial-time construction of a d -local function*

$$g : \{0, 1\}^n \rightarrow \{0, 1\}^{m'(n)}, \quad m'(n) \leq an,$$

with input locality at most D , which is $(2^{(\beta-\rho)n}, 2^{-(\beta-\rho)n})$ -one-way for all sufficiently large n . Here $a, D > 0$ depend only on d, ρ ; one may take

$$a = C_{d,\rho/2}, \quad D = \left\lceil \frac{2da}{\rho} \right\rceil,$$

where $C_{d,\rho/2}$ is the constant in Theorem 3.1. Moreover, for every constant $\alpha > 0$ there are a constant $c = c(d, \rho, \alpha) > 0$ and integers $\ell_n \in \{0, \dots, n\}$ such that

$$\Pr_{X \leftarrow \mathcal{U}_n} [\ell_n \leq r_g(X) \leq \ell_n + \alpha n] \geq 1 - 2^{-cn} \quad (11)$$

for all sufficiently large n . Thus g is $(\alpha, 2^{-cn})$ -typically almost regular. The same g works for every fixed $\alpha > 0$ and has the form

$$g(x) = f(R[x])_S,$$

where R fixes at most ρn coordinates and $|S| \leq an$. The construction requires neither the regularity advice ℓ_n nor the exceptional-mass exponent c , and preserves polynomial-time uniformity.

To prove Theorem 4.1, we combine the sparsification theorem with two lemmas. The first bounds input locality by fixing a small number of input coordinates. The second shows that a function with constant input locality is typically almost-regular. Their proofs are deferred to Subsections 4.1 and 4.2, respectively.

Lemma 4.2 (Input-locality reduction). *Fix constants $a > 0$ and $0 < \eta < 1$, an integer $d \geq 1$, and an integer $D \geq \max\{1, ad/\eta\}$. Every d -local map*

$$h : \{0, 1\}^n \rightarrow \{0, 1\}^m, \quad m \leq an,$$

has a set $I \subseteq [n]$ of at most ηn input coordinates such that, after any fixing of the coordinates in I , the restricted map still has output locality at most d and has input locality at most D . In particular, every such restricted map h_R satisfies $\text{Ham}(h_R(x), h_R(x')) \leq D \cdot \text{Ham}(x, x')$ for all $x, x' \in \{0, 1\}^n$, where $\text{Ham}(\cdot, \cdot)$ denotes Hamming distance. The set I is deterministically computable in polynomial time from the local description of h .

Lemma 4.3 (Input locality implies typical almost-regularity). *Fix an integer $D \geq 1$ and constant $a > 0$. Let*

$$g : \{0, 1\}^n \rightarrow \{0, 1\}^m, \quad m \leq an,$$

and suppose that g has input locality of D . For every constant $\alpha > 0$, there is a constant $c = c(a, D, \alpha) > 0$ and a real number q_n such that

$$\Pr_{X \leftarrow \mathcal{U}_n} [q_n \leq \iota_g(X) \leq q_n + \alpha n] \geq 1 - 2^{-cn}$$

for all sufficiently large n .

The upper bound an on the output length is redundant: constant input locality D implies that there are at most Dn nonconstant outputs.

Proof of Theorem 4.1. Apply Theorem 3.1 with restriction budget $\rho/2$. It gives a restriction R_0 fixing at most $\rho n/2$ coordinates and a set S of at most $an = C_{d,\rho/2}n$ original outputs such that

$$h(x) = f(R_0[x])_S$$

determines the full restricted output $f(R_0[x])$. Apply Lemma 4.2 to h with $\eta = \rho/2$ and $D = \lceil 2da/\rho \rceil$. Extend R_0 by fixing every still-free coordinate in the resulting set I to zero. Writing R for the combined restriction, define

$$g(x) = f(R[x])_S.$$

Then $|\text{dom } R| \leq \rho n$, the output length is at most an , the output locality is at most d , and the input locality is at most D . The exact output-determination property of sparsification remains valid after this further restriction.

For one-wayness, an inverter for g with success probability q gives an inverter for f with success probability at least $2^{-|\text{dom } R|}q$: given $y = f(X)$, run the inverter on y_S and apply R to its answer. On the event that X satisfies R , its challenge is distributed as $g(U_n)$, and a successful projected inversion is a full inversion by exact output determination. For an inverter of size at most $2^{(\beta-\rho)n}$, the resulting circuit has size at most

$$2^{(\beta-\rho)n} + \text{poly}(n, m) < 2^{\beta n}$$

for sufficiently large n . Thus

$$q \leq 2^{|\text{dom } R| - \beta n} \leq 2^{-(\beta-\rho)n}.$$

Finally, apply Lemma 4.3 with width $\alpha n/2$. Since for every $x \in \{0, 1\}^n$

$$r_g(x) = n - \iota_g(x),$$

the interval

$$q_n \leq \iota_g(X) \leq q_n + \alpha n/2$$

given by the lemma translates into

$$n - q_n - \alpha n/2 \leq r_g(X) \leq n - q_n.$$

Intersect this interval with $[0, n]$ and round its lower endpoint down to an integer ℓ_n . This increases its width by less than one, which is absorbed by the remaining $\alpha n/2$ slack for all sufficiently large n , proving (11).

The constants a, D depend only on d, ρ , so the resulting $c > 0$ depends only on d, ρ, α . All restriction and projection choices are deterministically polynomial-time constructible, and neither these choices nor g depend on α, ℓ_n , or c . $\square$

4.1 Proof of the input-locality reduction

For a local map with linear output length, the average number of outputs influenced by an input is constant. Fixing the inputs whose influence exceeds a sufficiently large constant therefore costs only a small fraction of the input coordinates. The proof of the main theorem charges this restriction together with the sparsification restriction when transferring one-wayness.

Proof of Lemma 4.2. Let $\deg(i)$ be the number of output functions of h that essentially depend on input coordinate i . Since there are at most an outputs and each depends on at most d inputs,

$$\sum_{i=1}^n \deg(i) \leq adn.$$

Set

$$I = \{i : \deg(i) > D\}.$$

Then, by an averaging argument,

$$|I| \leq \frac{ad}{D}n \leq \eta n.$$

After any fixing of the coordinates in I , each remaining input influences at most D outputs, and the essential support of each output can only decrease. The restricted map therefore has input locality at most D and output locality at most d . We retain the original n -bit domain, with the fixed coordinates regarded as unused inputs. Changing one input bit changes at most D output bits; changing differing input bits one at a time gives

$$\text{Ham}(h_R(x), h_R(x')) \leq D \text{Ham}(x, x').$$

The degrees and I can be computed directly from the essential supports in the local description, in deterministic polynomial time. $\square$

4.2 Proof that bounded input locality implies typical almost-regularity

The proof is based on the standard isoperimetric/blowing-up phenomenon for the Hamming cube. Related inequalities were used by Bogdanov and Rosen [17] to amplify the hardness of input-local one-way functions.

Proof of Lemma 4.3. Pad the output with zero coordinates to length $M = \lceil an \rceil$. This changes neither output probabilities nor Hamming distances. Let h_2 denote binary entropy. Choose a sufficiently small constant $\theta > 0$ such that

$$\frac{D\theta}{a} < \frac{1}{2}, \quad ah_2\left(\frac{D\theta}{a}\right) + \frac{\theta^2}{2\ln 2} < \alpha. \quad (12)$$

Set $\varepsilon = \exp(-\theta^2 n/2)$ and let $q = q_n$ be the smallest value of ι_g on $\text{Im}(g)$ for which

$$\Pr[\iota_g(X) \leq q] \geq \varepsilon, \quad X \leftarrow U_n.$$

Define

$$T = \{y \in \text{Im}(g) : \iota_g(y) \leq q\}, \quad A = g^{-1}(T) = \{x : \iota_g(x) \leq q\}.$$

Then

$$\Pr[X \in A] \geq \varepsilon, \quad \Pr[\iota_g(X) < q] < \varepsilon, \quad |T| \leq 2^q. \quad (13)$$

The last inequality follows since every $y \in T$ has probability $\Pr[g(X) = y] = 2^{-\iota_g(y)} \geq 2^{-q}$.

Blowing up the core. Let $F(x) = \text{Ham}(x, A)$ and $\mu = \mathbb{E}F(X)$. Changing one input bit changes F by at most one. The one-sided bounded-differences inequalities (McDiarmid's inequality) therefore give

$$\Pr[F(X) - \mu \leq -u] \leq e^{-2u^2/n}, \quad \Pr[F(X) - \mu \geq u] \leq e^{-2u^2/n} \quad (u > 0).$$

If $\mu > 0$, take $u = \mu$ in the first bound. Since $F(X) = 0$ exactly when $X \in A$,

$$\varepsilon \leq \Pr[X \in A] = \Pr[F(X) = 0] \leq e^{-2\mu^2/n},$$

and hence

$$\mu \leq \sqrt{\frac{n}{2} \ln \frac{1}{\varepsilon}} = \frac{\theta n}{2}.$$

The same conclusion is immediate if $\mu = 0$.

Let $t = \lceil \theta n \rceil$ and

$$A_t = \{x : \text{Ham}(x, A) \leq t\}.$$

The second bound now implies

$$\Pr[X \notin A_t] \leq \exp\left(-\frac{2(t - \mu)^2}{n}\right) \leq \varepsilon. \quad (14)$$

Bounding the enlarged image. If x differs from some $x' \in A$ in at most t input coordinates, then at most Dt output coordinates can change, since each input coordinate influences at most D outputs. Hence

$$g(A_t) \subseteq T_{Dt},$$

where T_{Dt} denotes the Hamming neighborhood of radius Dt around T . Letting

$$V = \sum_{j=0}^{\min\{M, Dt\}} \binom{M}{j},$$

we have

$$|g(A_t)| \leq |T|V \leq 2^q V. \quad (15)$$

In any set of at most $2^q V$ output strings, the total probability under $g(X)$ of elements y satisfying

$$\iota_g(y) > q + \log_2 V + \log_2(1/\varepsilon)$$

is at most ε . Combining this observation with (13) and (14) yields

$$\Pr [q \leq \iota_g(X) \leq q + \log_2 V + \log_2(1/\varepsilon)] \geq 1 - 3\varepsilon. \quad (16)$$

Finally, the Hamming-ball volume bound and (12) give

$$\frac{\log_2 V + \log_2(1/\varepsilon)}{n} \leq ah_2\left(\frac{D\theta}{a}\right) + \frac{\theta^2}{2\ln 2} + o(1) < \alpha$$

for all sufficiently large n . Since $3\varepsilon \leq 2^{-cn}$ for some constant $c > 0$, the lemma follows. $\square$

5 Constructing PRGs

The following (non-local) construction is identical to the one presented in [4] and it can be viewed as a simplified version of the HILL construction [29] (see also [23, Section 3.5.2]). At a high level, starting from a uniform input X and its image $g(X)$, the construction information-theoretically extracts randomness from the input conditioned on the output and from the output itself, and then computationally extracts additional hardcore pseudorandom bits from the input.

Construction 5.1 (Non-Local PRG construction). *Let $g : \{0, 1\}^n \rightarrow \{0, 1\}^m$ be a function, and let $s_1, s_2, s_3 = O(n)$ be integer parameters. Instantiate Theorem 2.7 three times as follows:*

$$H : \{0, 1\}^n \times \{0, 1\}^{t_1} \rightarrow \{0, 1\}^{s_1}, \quad C : \{0, 1\}^n \times \{0, 1\}^{t_2} \rightarrow \{0, 1\}^{s_2},$$

and

$$E : \{0, 1\}^{m+s_1} \times \{0, 1\}^{t_3} \rightarrow \{0, 1\}^{s_3}.$$

For a key w_i , write $H_{w_1}(x) = H(x, w_1)$, $C_{w_2}(x) = C(x, w_2)$, and $E_{w_3}(z) = E(z, w_3)$.

Define

$$G_g(x, w_1, w_2, w_3) = (w_1, w_2, w_3, E_{w_3}(g(x), H_{w_1}(x)), C_{w_2}(x)). \quad (17)$$

Denoting by N and M the input length and the output length, respectively, we have

$$N = n + t_1 + t_2 + t_3, \quad M = t_1 + t_2 + t_3 + s_2 + s_3.$$

In particular, the additive stretch is $s_2 + s_3 - n$.

Remark 5.2. The above construction applies the second hashing E_{w_3} to $(g(x), H_{w_1}(x))$. Alternatively, one may apply this final extractor only to $g(x)$, and output the first-hash value $H_{w_1}(x)$ separately. With a suitable split of the output lengths, the two extracted blocks together achieve the same linear stretch; we keep the present formulation for consistency with [4].

The original analysis of [4] applies to almost-regular OWFs. Below, we extend it to the setting of typically almost-regular OWFs. Since the construction evaluates the OWF only once, the exponentially small exceptional set is charged only at the end. (See Section B for a full proof.)

Theorem 5.3 (Typically almost-regular OWF gives a PRG). *Let $g : \{0, 1\}^n \rightarrow \{0, 1\}^m$, $m = O(n)$, be $(2^{bn}, 2^{-bn})$ -one-way for a positive rational constant $b > 0$, and set $\tau = \min \left\{ \frac{b}{24}, \frac{1}{12} \right\}$. Suppose that for some integer $\ell = \ell_n$ and some constant $c > 0$,*

$$\Pr_{X \leftarrow \mathcal{U}_n} [\ell \leq r_g(X) \leq \ell + \tau n] \geq 1 - 2^{-cn}. \quad (18)$$

Instantiate Construction 5.1 with

$$s_1 = \max\{0, \lfloor \ell - 3\tau n \rfloor\}, \quad s_2 = \lfloor 6\tau n \rfloor, \quad s_3 = \lfloor (1 - 5\tau)n \rfloor.$$

Then G_g is an exponentially secure PRG with additive stretch

$$s_2 + s_3 - n = \tau n - O(1).$$

More precisely, for some constant $\gamma = \gamma(b, c) > 0$ and all sufficiently large n , the generator is $(2^{\gamma n}, 2^{-\gamma n})$ -secure and efficiently constructible given ℓ_n as advice.

For our application, the rationality assumption on b is without loss of generality: we may decrease the hardness exponent to a positive rational constant before choosing the regularity width.

Remark 5.4 (Parameters and advice). *The construction needs two kinds of parameters. First, the constant τ only requires a fixed lower bound on the hardness exponent b . This choice is universal across all input lengths and therefore does not constitute length-dependent advice. Second, the first hash length s_1 depends on the lower endpoint ℓ_n of the regularity window. Thus Theorem 5.3, as stated, uses one integer of regularity advice at each input length. The exponent c of the exceptional mass is used only in the security analysis and is not needed by the construction. The corollary below removes the regularity advice by trying a constant number of candidate endpoints and combining all the resulting generators into a single function.*

The following corollary mimics [4, Corollary 5.3].

Corollary 5.5 (Local, advice-free PRG from typically almost-regular OWF). *Let*

$$g_n : \{0, 1\}^n \rightarrow \{0, 1\}^{m_n}, \quad m_n = O(n),$$

be a constant-locality function which is $(2^{bn}, 2^{-bn})$ -one-way for some positive rational constant $b > 0$, and set $\tau = \min \left\{ \frac{b}{24}, \frac{1}{12} \right\}$. Suppose that for some constant $c > 0$ and every sufficiently large n there is an integer ℓ_n such that

$$\Pr_{X \leftarrow \mathcal{U}_n} \left[\ell_n \leq r_{g_n}(X) \leq \ell_n + \frac{\tau n}{2} \right] \geq 1 - 2^{-cn}. \quad (19)$$

Then there exists an exponentially secure PRG in NC^0 with constant multiplicative stretch. In fact, the final generator can be taken to have locality four. Moreover, if g_n is efficiently computable, then so is the resulting PRG.

We emphasize that no length-dependent advice or collection index is required. Together with Theorem 4.1 we derive the following theorem (the PRG-part of the main theorem).

Theorem 5.6 (Bitwise-optimal PRG from bitwise-optimal OWF). *Suppose that for some constants $d \in \mathbb{N}$ and $\beta > 0$ there is a d -local, efficiently computable function*

$$f : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)},$$

that is $(2^{\beta n}, 2^{-\beta n})$ -one-way. Then there exist constants $\lambda, \gamma > 0$ and a four-local pseudorandom generator

$$G : \{0, 1\}^N \rightarrow \{0, 1\}^M,$$

where

$$N = \Theta(n), \quad M = O(N), \quad M \geq (1 + \lambda)N,$$

such that G is $(2^{\gamma N}, 2^{-\gamma N})$ -secure.

5.1 Proof of Corollary 5.5

We first explain the construction assuming that a valid endpoint is known, and then remove this advice.

Localizing one candidate. Fix an endpoint a for which

$$\Pr[a \leq r_{g_n}(X) \leq a + \tau n] \geq 1 - 2^{-\Omega(n)}. \quad (20)$$

Apply Theorem 5.3 with $\ell = a$ and obtain the (nonlocal) generator

$$G_a(x, w_1, w_2, w_3).$$

We localize G_a exactly as in the proof of [4, Corollary 5.3]. First view $z = g_n(x)$ as an independent input. The two-level composition guarantee of Theorem 2.7 gives a three-local perfect randomized encoding of

$$(x, z, w_1, w_3) \mapsto (w_1, w_3, E_{w_3}(z, H_{w_1}(x)))$$

with $O(n)$ randomness complexity and with both keys copied verbatim to the output. The single-hash encoding guarantee of the same theorem gives a three-local perfect randomized encoding of $(x, w_2) \mapsto (w_2, C_{w_2}(x))$, again with $O(n)$ randomness complexity and with the key copied verbatim to the output. Concatenating these two encodings using independent encoding coins, and reordering the output coordinates, yields a perfect randomized encoding of

$$(x, z, w_1, w_2, w_3) \mapsto (w_1, w_2, w_3, E_{w_3}(z, H_{w_1}(x)), C_{w_2}(x)),$$

by the standard concatenation property of perfect randomized encodings [7]. The resulting encoding has $O(n)$ randomness and output complexity and is three-local jointly in x, z, w_1, w_2, w_3 and all encoding coins.

Substitute $z = g_n(x)$. Since each coordinate of g_n has constant locality, the substitution increases the locality by only a constant factor. By the substitution lemma of [10], we obtain a constant-local perfect randomized encoding

$$\widehat{G}_a(x, w_1, w_2, w_3; r)$$

of G_a whose seed length and output length are both $\Theta(n)$. Balanced simulation implies that a perfect randomized encoding of a PRG is again a PRG, and the length-preserving property of the encoding preserves the additive stretch exactly [7]. Hence $\widehat{G}_a$ is an exponentially secure constant-locality PRG with additive stretch $\Omega(n)$.

Removing the regularity advice. We follow the argument of [4, Corollary 5.3] (which is a variant of [29]). Set $u_n = \lfloor \tau n/4 \rfloor$ and try every endpoint in

$$\mathcal{L}_n = \{0, u_n, 2u_n, \dots, \lfloor n/u_n \rfloor u_n\} \cup \{n\}.$$

For some $a \in \mathcal{L}_n$, we have $a \leq \ell_n < a + u_n$, so $[\ell_n, \ell_n + \tau n/2] \subseteq [a, a + \tau n]$. Thus, by (19) and the preceding localization argument, at least one of the resulting constant-locality candidates is exponentially secure. All candidates have the same additive stretch. Padding shorter seeds with fresh bits copied to the output, and repeating candidates if necessary, gives a fixed number $J = O_\tau(1)$ of maps

$$P_j : \{0, 1\}^N \rightarrow \{0, 1\}^{N+\Delta}, \quad N = \Theta(n), \quad \Delta = \Omega(n).$$

By standard stretch amplification using a constant number of self-compositions at the same seed length, followed by truncation, we obtain maps $Q_j : \{0, 1\}^N \rightarrow \{0, 1\}^{CN}$ for any fixed integer $C > J$, preserving

constant locality and exponential security of each secure candidate. Here each iteration feeds back N output bits and retains the remaining bits, so every invocation uses the same base parameter n . On independent seeds, define

$$Q(s^{(1)}, \dots, s^{(J)}) = \bigoplus_{j=1}^J Q_j(s^{(j)}).$$

Replacing one secure summand by an independent uniform string makes the XOR exactly uniform. Hence Q is exponentially secure, has constant locality, and stretches JN bits to CN bits.

Finally, as in [4, Corollary 5.3], a linear-overhead four-local perfect randomized encoding preserves additive stretch and exponential security, yielding the claimed PRG. All candidates are constructed and combined without identifying a secure one, so no regularity advice is needed and P-uniformity is preserved.⁴ $\square$

6 Universal one-way hash functions

We begin with a non-local construction of UOWHF that employs two pairwise independent hash functions. The first hash separates points within a typical fiber, and the second provides the final compression. This follows the hash-then-compress approach originating with Naor and Yung [38] and further developed by De Santis and Yung [22].

Construction 6.1 (The two-hash UOWHF construction). *Let $g : \{0, 1\}^n \rightarrow \{0, 1\}^m$, where $m = O(n)$, and let $s_1, s_2 = O(n)$ be positive integer parameters, with $s_2 < n$. Instantiate the hashing theorem (Thm. 2.7) from the preliminaries twice to obtain*

$$H : \{0, 1\}^n \times \{0, 1\}^{t_1} \rightarrow \{0, 1\}^{s_1}, \quad E : \{0, 1\}^{m+s_1} \times \{0, 1\}^{t_2} \rightarrow \{0, 1\}^{s_2},$$

where $t_1, t_2 = O(n)$. When necessary, pad a hash input with fixed zeroes to length at least n before applying Thm. 2.7; the displayed maps denote the resulting restrictions to their natural domains. The public key is $k = (a, w_1, w_2)$, with all components sampled independently and uniformly and $a \in \{0, 1\}^n$. Define

$$\mathsf{U}_k^g(x) = E_{w_2}(g(x \oplus a), H_{w_1}(x \oplus a)). \quad (21)$$

Equivalently, the two stages, before the input mask, are

$$F_{w_1}(x) = (g(x), H_{w_1}(x)), \quad E_{w_2}(F_{w_1}(x)).$$

Thus U^g hashes n message bits to s_2 bits using an $(n + t_1 + t_2)$ -bit public key. The mask converts random-target security into the standard notion in which the target is chosen before the public key is revealed. The public key is not part of the digest.

We analyze the construction for the case of exponentially-strong typically almost regular one-way functions. Closely related variants appear in the literature. Most notably, the cases of efficient known-regularity and almost-injectivity were handled by Yu, Gu, Li, and Weng [48]. The following theorem is a quantitative adaptation, rather than a verbatim application of these results: regularity holds only outside an exponentially small exceptional set, the allowed fiber-size ratio is exponential, and we require exponential security with linear shrinkage for the particular key-samplable hash families in the preliminaries.⁵ These are the points addressed by the regularization and collision-to-inversion arguments.

⁴For every initial input length n , the theorem provides a PRG of input length $n' \in [n, Bn]$ for some constant $B \geq 1$, and therefore may not cover all input lengths. This can be easily fixed via a standard padding argument. Concretely, for a requested length L , use $n = \lfloor L/B \rfloor$, apply the construction to its required input coordinates, and copy the remaining input bits to the output. Multiplicative stretch remains constant.

⁵In fact, the key-samplability property is only required for the external hashing E .

Theorem 6.2 (Typically almost-regular OWF gives a UOWHF). *Let $g : \{0, 1\}^n \rightarrow \{0, 1\}^m$, with $m = O(n)$, be $(2^{bn}, 2^{-bn})$ -one-way for a constant $b \in (0, 1)$. Fix $c > 0$ and positive rational constants α, θ satisfying*

$$0 < \alpha \leq b/16, \quad 0 < \theta \leq \frac{1}{16} \min\{\alpha, c\}.$$

Suppose that g is typically almost regular in the sense that, for some integer $\ell = \ell_n \in \{0, \dots, n\}$,

$$\Pr_{X \leftarrow \mathcal{U}_n} [\ell \leq r_g(X) \leq \ell + \alpha n] \geq 1 - 2^{-cn}. \quad (22)$$

Instantiate Construction 6.1 with

$$s_1 = \lceil \ell + 2\alpha n \rceil, \quad s_2 = n - \lceil \theta n \rceil. \quad (23)$$

Then $\mathcal{U}^g$ is a UOWHF with linear key length and additive shrinkage $\lceil \theta n \rceil$ which is efficiently constructible given ℓ_n as advice. More precisely, for some constant $\gamma > 0$ and all sufficiently large n , it has target-collision success probability at most $2^{-\gamma n}$ against adversaries of size at most $2^{\gamma n}$.

The security analysis requires pairwise independence of both H and E , and key-samplability only for E . The proof is given in Section C.

Remark 6.3 (Parameters and regularity advice). *The only length-dependent parameter in (23) is the regularity advice ℓ_n . The constants α and θ are fixed once and for all, with α chosen sufficiently small relative to the one-wayness exponent b , and θ sufficiently small relative to α and the exceptional-mass exponent guaranteed by Theorem 4.1. Thus, they do not constitute advice. In particular, no relation between the exceptional-mass exponent and α is required. The corollary below removes ℓ_n by a constant grid.*

The following corollary is the UOWHF analogue of the PRG corollary (Corollary 5.5): first localize each candidate, and then remove its regularity advice by combining a constant number of candidates. For UOWHFs we concatenate the candidate digests on the *same message*, after increasing their compression sufficiently.

Corollary 6.4 (Local, advice-free UOWHF from typically almost-regular OWF). *Let $g_n : \{0, 1\}^n \rightarrow \{0, 1\}^{m_n}$, with $m_n = O(n)$, be a d -local, $(2^{bn}, 2^{-bn})$ -one-way function, for constants $d, b > 0$. Let $c > 0$ and a rational $0 < \alpha \leq \min\{b/16, 1/16\}$ be fixed constants. Suppose that for every sufficiently large n there is an integer $\ell_n \in \{0, \dots, n\}$ for which*

$$\Pr_{X \leftarrow \mathcal{U}_n} [\ell_n \leq r_{g_n}(X) \leq \ell_n + \alpha n/2] \geq 1 - 2^{-cn}. \quad (24)$$

Then there is a UOWHF collection with a four-local evaluation function $(x, k) \mapsto h_k(x)$, message length $N_n = \Theta(n)$, key length $O(n)$, constant multiplicative shrinkage, and exponential target-collision security in N_n . The transformation preserves P -uniformity.

The proof of the corollary is deferred to Section 6.1. Together with Theorem 4.1 we derive the following theorem (the UOWHF-part of the main theorem).

Theorem 6.5 (Bitwise-optimal UOWHF from bitwise-optimal OWF). *Suppose that for some constants $d \in \mathbb{N}$ and $\beta > 0$ there is a d -local, efficiently computable function*

$$f : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}, \quad m(n) = \text{poly}(n),$$

that is $(2^{\beta n}, 2^{-\beta n})$ -one-way. Then there exist constants $\mu, \gamma > 0$ and an efficiently computable collection

$$\mathcal{H} = \{h_k : \{0, 1\}^L \rightarrow \{0, 1\}^R\}_{k \in \{0, 1\}^{t(L)}},$$

where

$$L = \Theta(n), \quad t(L) = O(L), \quad R \leq (1 - \mu)L,$$

such that the evaluation function $(x, k) \mapsto h_k(x)$ is four-local and $\mathcal{H}$ is $(2^{\gamma L}, 2^{-\gamma L})$ -secure against target collisions.

6.1 Proof of Corollary 6.4

Fix a positive rational $\theta \leq \min\{\alpha, c\}/16$. We first localize a single candidate; this operation is well-defined even when its guessed endpoint is invalid.

Localizing one candidate. Fix a candidate endpoint ℓ and the lengths in (23), and let

$$\mathsf{U}^\ell(k, x) = (k, \mathsf{U}_k^\ell(x))$$

denote the corresponding “extended” UOWHF evaluation map, where $k = (a, w_1, w_2)$ and where the key is appended to the output. Viewing $z = g_n(u)$ for the moment as an independent input, apply the two-level encoding guarantee of Theorem 2.7 to obtain a three-local perfect randomized encoding of

$$(u, z, w_1, w_2) \mapsto (w_1, w_2, E_{w_2}(z, H_{w_1}(u)))$$

with $O(n)$ randomness complexity and with both keys copied verbatim to the output.

Now substitute $u = x \oplus a$ and $z = g_n(x \oplus a)$, and append a verbatim to the output. Since g_n has constant locality, the substitution lemma [10] yields a constant-local perfect randomized encoding

$$\widehat{\mathsf{U}}^\ell(k, x; r)$$

of $\mathsf{U}^\ell(k, x)$ using $\rho_\ell(n) = O(n)$ random bits. Write

$$\widehat{\mathsf{U}}^\ell(k, x; r) = (k, \widehat{\mathsf{U}}_k^\ell(x, r)),$$

where the first component is the verbatim copy of the key. We regard the encoding coins r as additional message coordinates and take $\widehat{\mathsf{U}}_k^\ell(x, r)$ as the localized hash evaluation.

The resulting evaluation map has constant locality, including dependence on the public-key bits, and key length $O(n)$. Moreover, preservation of additive length gives the common shrinkage

$$\Delta_n = n - s_2 = \lceil \theta n \rceil. \quad (25)$$

Following [7], we verify that target-collision resistance is preserved. Perfect correctness maps any collision of the encoding to equal unencoded digests. Moreover, the unique-randomness property implies that, for fixed k, x , the map

$$r \mapsto \widehat{\mathsf{U}}_k^\ell(x, r)$$

is injective. Hence a distinct collision $(x', r') \neq (x, r)$ must satisfy $x' \neq x$, and therefore yields a target collision for U_k^ℓ . Since the adversary chooses its target (x, r) before seeing k , forwarding x respects the target-before-key convention. The reduction has only polynomial overhead. Thus every secure nonlocal candidate yields a secure localized candidate, with exponential security in its $\Theta(n)$ -bit message length.

Removing the regularity advice. Set $u_n = \lfloor \alpha n / 4 \rfloor$ and try every endpoint in

$$\mathcal{L}_n = \{0, u_n, 2u_n, \dots, \lfloor n/u_n \rfloor u_n\} \cup \{n\}.$$

For some $\ell \in \mathcal{L}_n$, we have $\ell \leq \ell_n < \ell + u_n$, so $[\ell_n, \ell_n + \alpha n / 2] \subseteq [\ell, \ell + \alpha n]$. Thus at least one candidate is secure by Theorem 6.2. Localize every candidate, pad shorter messages with bits copied to the digest, and repeat candidates if necessary. Since the additive shrinkage is the same for all candidates, this gives a fixed number $J = O_\alpha(1)$ of maps

$$P_{j,k} : \{0, 1\}^N \rightarrow \{0, 1\}^M, \quad M = N - \Delta_n, \quad N = \Theta(n), \quad \Delta_n = \lceil \theta n \rceil, \quad (26)$$

at least one of which is exponentially secure at each length.

By standard compression amplification using constant-length chains with independent keys, obtain maps $Q_{j,k_j} : \{0,1\}^L \rightarrow \{0,1\}^M$, where

$$L = M + t\Delta_n \geq JM + \Delta_n \quad (27)$$

for a sufficiently large constant t . Explicitly, parse $v = (v_0, b_1, \dots, b_t)$ with $|v_0| = M$, $|b_i| = \Delta_n$, and set

$$v_i = P_{j,k_{j,i}}(v_{i-1}, b_i), \quad Q_{j,k_j}(v) = v_t. \quad (28)$$

Every invocation uses the same base parameter n and input length N . A collision in the chain yields a collision at some link. Guessing that link loses at most a factor t ; the reduction samples the preceding keys before committing to the link's target, preserving the target-before-key order. Thus constant locality, linear key length, and exponential security of a secure candidate are preserved.

On the same message and independent key tuples, define

$$Q_{k_1, \dots, k_J}(v) = (Q_{1,k_1}(v), \dots, Q_{J,k_J}(v)). \quad (29)$$

Any collision is a collision for a secure component, so Q is exponentially secure. It maps $L = \Theta(n)$ bits to JM bits, has $\Omega(n)$ shrinkage, constant locality, and an $O(n)$ -bit key. Finally, we apply a final linear-overhead four-local perfect encoding [7]. The final encoding can be applied to the joint evaluation map, with the public key treated as part of the input and copied to the output. Consequently, the resulting UOWHF (with the public key omitted from the output) has a four-local evaluation function

$$(k, x) \mapsto h_k(x),$$

jointly in the key and message bits. Additive shrinkage and target-collision security are preserved by the same correctness and unique-randomness argument above. All candidates are constructed and combined without identifying a secure one, so no regularity advice is needed and P-uniformity is preserved. A secure index is used only in the reduction, in accordance with our nonuniform-adversary convention.⁶ $\square$

References

- [1] Omar Alrabiah, Eshan Chattopadhyay, Jesse Goodman, Xin Li, and João Ribeiro. Low-degree polynomials extract from local sources. In *49th International Colloquium on Automata, Languages, and Programming (ICALP 2022)*, volume 229 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 10:1–10:20, 2022. doi:10.4230/LIPIcs.ICALP.2022.10.
- [2] Benny Applebaum. Pseudorandom generators with long stretch and low locality from random local one-way functions. *SIAM Journal on Computing*, 42(5):2008–2037, 2013. Preliminary version appeared in STOC 2012. doi:10.1137/120884857.
- [3] Benny Applebaum. Cryptographic hardness of random local functions: Survey. *Computational Complexity*, 25(3):667–722, 2016. doi:10.1007/s00037-015-0121-8.
- [4] Benny Applebaum. Exponentially-hard Gap-CSP and local PRG via local hardcore functions. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 836–847. IEEE Computer Society, 2017. See a full version in ECCC, Report TR17-063, <https://eccc.weizmann.ac.il/report/2017/063/>. Section, theorem, lemma, definition, corollary, and footnote locators in this manuscript refer to that report. doi:10.1109/FOCS.2017.82.

⁶For every initial input length n , the theorem provides a UOWHF of input length $n' \in [n, Bn]$ for some constant $B \geq 1$, and therefore may not cover all input lengths. This can be easily fixed via a standard padding argument as in Footnote 4.

- [5] Benny Applebaum, Naama Haramaty-Krasne, Yuval Ishai, Eyal Kushilevitz, and Vinod Vaikuntanathan. Low-complexity cryptographic hash functions. In *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *LIPIcs*, pages 7:1–7:31. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017. Full version: IACR Cryptology ePrint Archive, Report 2017/036. doi:10.4230/LIPIcs.ITCS.2017.7.
- [6] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Computationally private randomizing polynomials and their applications. *Computational Complexity*, 15(2):115–162, 2006. Preliminary version appeared in the 20th IEEE Conference on Computational Complexity (CCC 2005). doi:10.1007/s00037-006-0211-8.
- [7] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography in NC^0 . *SIAM Journal on Computing*, 36(4):845–888, 2006. doi:10.1137/S0097539705446950.
- [8] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. On pseudorandom generators with linear stretch in NC^0 . *Computational Complexity*, 17(1):38–69, 2008. Preliminary version appeared in RANDOM 2006. doi:10.1007/s00037-007-0237-6.
- [9] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography with constant input locality. *Journal of Cryptology*, 22(4):429–469, 2009. doi:10.1007/s00145-009-9039-0.
- [10] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. How to garble arithmetic circuits. *SIAM Journal on Computing*, 43(2):905–929, 2014. doi:10.1137/120875193.
- [11] Benny Applebaum and Eliran Kachlon. Sampling graphs without forbidden subgraphs and unbalanced expanders with negligible error. *SIAM Journal on Computing*, 52(6):1321–1368, 2023. Preliminary version appeared in FOCS 2019. doi:10.1137/22M1484134.
- [12] Benny Applebaum and Yoni Moses. Locally computable UOWHF with linear shrinkage. *Journal of Cryptology*, 30(3):672–698, 2017. Preliminary version appeared in EUROCRYPT 2013. doi:10.1007/s00145-016-9232-x.
- [13] Benny Applebaum and Yoni Moses. Correction: Locally computable UOWHF with linear shrinkage. *Journal of Cryptology*, 36(2), 2023. doi:10.1007/s00145-023-09443-9.
- [14] Benny Applebaum and Pavel Raykov. Fast pseudorandom functions based on expander graphs. In *Theory of Cryptography—TCC 2016-B, Part I*, volume 9985 of *Lecture Notes in Computer Science*, pages 27–56. Springer, 2016. Full version: IACR Cryptology ePrint Archive, Report 2016/813. doi:10.1007/978-3-662-53641-4_2.
- [15] Joshua Baron, Yuval Ishai, and Rafail Ostrovsky. On linear-size pseudorandom generators and hardcore functions. *Theoretical Computer Science*, 554:50–63, 2014. doi:10.1016/j.tcs.2014.06.013.
- [16] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM Journal on Computing*, 13(4):850–864, 1984. Preliminary version appeared in FOCS 1982. doi:10.1137/0213053.
- [17] Andrej Bogdanov and Alon Rosen. Input locality and hardness amplification. *Journal of Cryptology*, 26(1):144–171, 2013. See a full version in ECCC, Report TR11-012, <https://eccc.weizmann.ac.il/report/2011/012/>. doi:10.1007/s00145-011-9117-y.

- [18] Andrej Bogdanov, Alon Rosen, and Kel Zin Tan. Sample efficient search to decision for k LIN. In *Advances in Cryptology—CRYPTO 2025, Part I*, volume 16000 of *Lecture Notes in Computer Science*, pages 203–220. Springer, 2025. Full version: IACR Cryptology ePrint Archive, Report 2025/1469. doi:10.1007/978-3-032-01855-7_7.
- [19] Andrej Bogdanov, Manuel Sabin, and Prashant Nalini Vasudevan. XOR codes and sparse learning parity with noise. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2019)*, pages 986–1004. SIAM, 2019. doi:10.1137/1.9781611975482.61.
- [20] Yi-Hsiu Chen, Mika Göös, Salil P. Vadhan, and Jiapeng Zhang. A tight lower bound for entropy flattening. In *33rd Computational Complexity Conference (CCC 2018)*, volume 102 of *LIPICs*, pages 23:1–23:28. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018. doi:10.4230/LIPICs.CCC.2018.23.
- [21] Anindya De and Thomas Watson. Extractors and lower bounds for locally samplable sources. *ACM Transactions on Computation Theory*, 4(1), 2012. Preliminary version appeared in RANDOM 2011. doi:10.1145/2141938.2141941.
- [22] Alfredo De Santis and Moti Yung. On the design of provably-secure cryptographic hash functions. In Ivan Bjerre Damgård, editor, *Advances in Cryptology—EUROCRYPT ’90*, volume 473 of *Lecture Notes in Computer Science*, pages 412–431. Springer, 1991. doi:10.1007/3-540-46877-3_37.
- [23] Oded Goldreich. *Foundations of Cryptography: Volume 1, Basic Tools*. Cambridge University Press, 2001. doi:10.1017/CBO9780511546891.
- [24] Oded Goldreich, Hugo Krawczyk, and Michael Luby. On the existence of pseudorandom generators. *SIAM Journal on Computing*, 22(6):1163–1175, 1993. Preliminary version appeared in FOCS 1988. doi:10.1137/0222069.
- [25] Oded Goldreich and Leonid A. Levin. A hard-core predicate for all one-way functions. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing (STOC 1989)*, pages 25–32. ACM, 1989. doi:10.1145/73007.73010.
- [26] Oded Goldreich, Ronitt Rubinfeld, and Madhu Sudan. Learning polynomials with queries: The highly noisy case. *SIAM Journal on Discrete Mathematics*, 13(4):535–570, 2000. doi:10.1137/S0895480198344540.
- [27] Iftach Haitner, Thomas Holenstein, Omer Reingold, Salil Vadhan, and Hoeteck Wee. Universal one-way hash functions via inaccessible entropy. In *Advances in Cryptology—EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 616–637. Springer, 2010. doi:10.1007/978-3-642-13190-5_31.
- [28] Iftach Haitner, Omer Reingold, and Salil Vadhan. Efficiency improvements in constructing pseudorandom generators from one-way functions. *SIAM Journal on Computing*, 42(3):1405–1430, 2013. Preliminary version appeared in STOC 2010. doi:10.1137/100814421.
- [29] 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. doi:10.1137/S0097539793244708.
- [30] Thomas Holenstein, Ueli Maurer, and Johan Sjödin. Complete classification of bilinear hard-core functions. In Matthew Franklin, editor, *Advances in Cryptology—CRYPTO 2004*, volume 3152 of *Lecture*

- Notes in Computer Science*, pages 73–91. Springer, 2004. doi:10.1007/978-3-540-28628-8_5.
- [31] Jordan Horacek, Chin Ho Lee, Igor Shinkar, Emanuele Viola, and Renfei Zhou. Local samplers for product distributions. In *53rd International Colloquium on Automata, Languages, and Programming (ICALP 2026)*, Leibniz International Proceedings in Informatics (LIPIcs), pages 109:1–109:23, 2026. doi:10.4230/LIPIcs.ICALP.2026.109.
 - [32] Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001. doi:10.1006/jcss.2001.1774.
 - [33] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Cryptography with constant computational overhead. In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing (STOC)*, pages 433–442. Association for Computing Machinery, 2008. doi:10.1145/1374376.1374438.
 - [34] Jonathan Katz and Chiu-Yuen Koo. On constructing universal one-way hash functions from arbitrary one-way functions. *IACR Cryptology ePrint Archive*, 2005(328), 2005. URL: <https://eprint.iacr.org/2005/328>.
 - [35] Xinyu Mao, Noam Mazon, and Jiapeng Zhang. Non-adaptive universal one-way hash functions from arbitrary one-way functions. In *Advances in Cryptology—EUROCRYPT 2023*, volume 14007 of *Lecture Notes in Computer Science*, pages 502–531. Springer, 2023. Full version: IACR Cryptology ePrint Archive, Report 2022/431. doi:10.1007/978-3-031-30634-1_17.
 - [36] Noam Mazon and Jiapeng Zhang. Simple constructions from (almost) regular one-way functions. *Journal of Cryptology*, 37(2):25, 2024. Preliminary version appeared in TCC 2021. doi:10.1007/s00145-024-09507-4.
 - [37] Elchanan Mossel, Amir Shpilka, and Luca Trevisan. On ϵ -biased generators in NC^0 . *Random Structures & Algorithms*, 29(1):56–81, 2006. doi:10.1002/rsa.20112.
 - [38] Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing (STOC)*, pages 33–43. Association for Computing Machinery, 1989. doi:10.1145/73007.73011.
 - [39] John Rompel. One-way functions are necessary and sufficient for secure signatures. In *Proceedings of the Twenty-Second Annual ACM Symposium on Theory of Computing (STOC 1990)*, pages 387–394. ACM, 1990. doi:10.1145/100216.100269.
 - [40] Victor Shoup. New algorithms for finding irreducible polynomials over finite fields. *Mathematics of Computation*, 54(189):435–447, 1990. doi:10.1090/S0025-5718-1990-0993933-0.
 - [41] Kel Zin Tan and Prashant Nalini Vasudevan. Improved search-to-decision reduction for random local functions. In *Advances in Cryptology—EUROCRYPT 2026, Part V*, volume 16545 of *Lecture Notes in Computer Science*, pages 214–244. Springer, 2026. Full version: ECCC Report TR25-139. doi:10.1007/978-3-032-25330-9_8.
 - [42] Salil P. Vadhan and Colin Jia Zheng. Characterizing pseudoentropy and simplifying pseudorandom generator constructions. In *Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC 2012)*, pages 817–836. ACM, 2012. Full version: ECCC Report TR11-141. doi:10.1145/2213977.2214051.

- [43] Emanuele Viola. The complexity of distributions. *SIAM Journal on Computing*, 41(1):191–218, 2012. Preliminary version appeared in FOCS 2010. doi:10.1137/100814998.
- [44] Emanuele Viola. Extractors for circuit sources. *SIAM Journal on Computing*, 43(2):655–672, 2014. Preliminary version appeared in FOCS 2011. doi:10.1137/11085983X.
- [45] Emanuele Viola. Sampling lower bounds: Boolean average-case and permutations. *SIAM Journal on Computing*, 49(1):119–137, 2020. doi:10.1137/18M1198405.
- [46] Emanuele Viola. New sampling lower bounds via the separator. In *38th Computational Complexity Conference (CCC 2023)*, volume 264 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 26:1–26:23, 2023. doi:10.4230/LIPIcs.CCC.2023.26.
- [47] Andrew C. Yao. Theory and applications of trapdoor functions. In *23rd Annual Symposium on Foundations of Computer Science (FOCS 1982)*, pages 80–91. IEEE, 1982. doi:10.1109/SFCS.1982.45.
- [48] Yu Yu, Dawu Gu, Xiangxue Li, and Jian Weng. (almost) optimal constructions of UOWHFs from 1-to-1, regular one-way functions and beyond. In Rosario Gennaro and Matthew Robshaw, editors, *Advances in Cryptology—CRYPTO 2015, Part II*, volume 9216 of *Lecture Notes in Computer Science*, pages 209–229. Springer, 2015. See a full version in the IACR Cryptology ePrint Archive, Report 2014/393, <https://eprint.iacr.org/2014/393>. Full-version locators refer to this report. doi:10.1007/978-3-662-48000-7_11.

A Compressing local sources: Proof of Corollary 3.3

The case $d = 1$ is immediate, so assume $d \geq 2$. We modify the probabilistic construction in the proof of the sparsification theorem. Roughly, that construction produces a random restriction R , consisting of an initial random restriction followed by a repair step, under which exact reconstruction holds. By randomizing the repair assignments, we ensure that the expected entropy loss is at most the expected number of fixed coordinates. The remaining issue is to find a single restriction that simultaneously respects the ρn fixing budget and loses at most ρn bits of entropy. To this end, we introduce additional parameter slack: by taking p smaller and increasing K , we reduce the expected number of fixed coordinates to $O(\rho^2 n)$. A weighted averaging argument then yields a restriction satisfying both requirements. We proceed with the formal details.

A smaller expected fixing budget. Let $\delta = \frac{\rho^2}{1+\rho}$. Choose the initial restriction density p and the petal-cover threshold K so that the fixing-cost bound $Z(R_0)$ from that proof satisfies

$$\mathbb{E}Z(R_0) \leq \delta n.$$

Concretely, take $p \leq \delta/2$ and then choose K sufficiently large that

$$dL_d B_K e^{-\kappa K} \leq \delta/2.$$

As before, this is possible because B_K grows polynomially in K , whereas the failure bound decays exponentially. The candidate family still has size $O_{d,\rho}(n)$.

Randomized repairs. Given the initial restriction R_0 , let J be the union of all still-free coordinates in the cores whose designated exposure events failed under R_0 . Instead of fixing these coordinates to zero, assign the coordinates in J independent uniform bits, obtaining R . The set J is determined by R_0 before these new values are chosen.

The exposure and local-determination arguments are unchanged: every failed core becomes fully fixed, and further restrictions preserve earlier exposures. Thus every outcome admits the required projection and local reconstruction. Writing

$$c(R) = |\text{dom}(R)|, \quad h(R) = H(f(R[U_n])),$$

we also retain the bound

$$c(R) \leq Z(R_0), \quad \mathbb{E}c(R) \leq \delta n.$$

Expected entropy loss. Let X be uniform over $\{0, 1\}^n$, and write $T = \text{dom}(R_0)$. Since the initial fixing on T is uniform,

$$\mathbb{E}_{R_0} h(R_0) = \mathbb{E}_T H(Y \mid X_T) \geq H(Y) - \mathbb{E}|T|.$$

For each fixed R_0 , the additional set J is fixed and its assigned values are uniform. The same conditional-entropy inequality therefore gives

$$\mathbb{E}[h(R) \mid R_0] \geq h(R_0) - |J|.$$

Since $c(R) = |T| + |J|$, averaging yields

$$\mathbb{E}h(R) \geq H(Y) - \mathbb{E}c(R).$$

Simultaneous control of fixing cost and entropy. Consider the potential

$$\Phi(R) = h(R) - \frac{c(R)}{\rho}.$$

The preceding bounds imply

$$\mathbb{E}\Phi(R) \geq H(Y) - \left(1 + \frac{1}{\rho}\right) \mathbb{E}c(R) \geq H(Y) - \left(1 + \frac{1}{\rho}\right) \delta n = H(Y) - \rho n.$$

Hence some outcome satisfies $\Phi(R) \geq H(Y) - \rho n$. Since $c(R) \geq 0$, this immediately gives

$$h(R) \geq H(Y) - \rho n.$$

Moreover, the restricted source is generated by only $n - c(R)$ independent bits, so $h(R) \leq n - c(R)$. Consequently,

$$H(Y) - \rho n \leq \Phi(R) \leq n - \left(1 + \frac{1}{\rho}\right) c(R),$$

and therefore

$$c(R) \leq \frac{\rho}{1 + \rho} ((1 + \rho)n - H(Y)) \leq \rho n.$$

For this outcome, apply the unchanged output-retention and local-reconstruction steps of the sparsification proof. Finally, Y_R and $(Y_R)_S$ determine one another, and hence have the same entropy. $\square$

B From typically almost-regular OWF to PRG: Proof of Thm. 5.3

Before proving the theorem we will need two technical lemmas.

First, the following simple observation (that will be also useful in the UOWHF construction) asserts that if, conditioned on $g(X) = y$, the input X is uniform over a set of size at least 2^ℓ , then revealing s bits obtained by pairwise-independent hashing of X cannot substantially increase the probability of inverting g . Importantly, the bound remains useful even when $s > \ell$. The proof is similar to the collision-based proof of the leftover hash lemma [29]; a weaker bound also follows from the leftover hash lemma in a black-box manner.

Lemma B.1 (Hash leakage preserves one-wayness). *Let X be a distribution over $\{0, 1\}^n$, let $Y = g(X)$, and suppose that for every $y \in \text{Supp}(Y)$, the conditional distribution $X \mid Y = y$ is uniform over a set of size at least 2^ℓ . Let*

$$H : \{0, 1\}^n \times \mathcal{W} \rightarrow \{0, 1\}^s$$

be pairwise independent, and let $W \leftarrow \mathcal{W}$ be uniform and independent of X .

Fix any probabilistic algorithm A , and let

$$q_0 = \Pr[g(A(Y, W, Z)) = Y], \quad Z \leftarrow U_s$$

where W, Z are independent of X, Y . Then

$$\Pr[g(A(Y, W, H_W(X))) = Y] \leq q_0 + \sqrt{q_0 2^{s-\ell}}. \quad (*)$$

Consequently, if inversion of g from Y has success probability at most ε for the relevant complexity bound, then revealing $W, H_W(X)$ increases the inversion probability to at most

$$\varepsilon + \sqrt{\varepsilon 2^{s-\ell}},$$

up to the polynomial overhead required to sample W, Z . In particular, the same bound applies to inversion of

$$F_W(x) = (g(x), H_W(x)).$$

Proof. For fixed y, w , let

$$P_{y,w}(z) = \Pr[H_w(X) = z \mid Y = y], \quad \text{and} \quad a(y, w, z) = \Pr[g(A(y, w, z)) = y],$$

where the probability is over the internal randomness of A . Then

$$q_0 = \mathbb{E}_{Y,W} 2^{-s} \sum_z a(Y, W, z).$$

If $K_y = |\text{Supp}(X \mid Y = y)|$, pairwise independence gives

$$\mathbb{E}_W \sum_z (P_{y,W}(z) - 2^{-s})^2 = \frac{1 - 2^{-s}}{K_y} \leq 2^{-\ell}.$$

Therefore, by Cauchy–Schwarz and $0 \leq a \leq 1$,

$$\begin{aligned} & \Pr[g(A(Y, W, H_W(X))) = Y] - q_0 \\ &= \mathbb{E}_{Y,W} \sum_z (P_{Y,W}(z) - 2^{-s}) a(Y, W, z) \\ &\leq \sqrt{\mathbb{E}_{Y,W} \sum_z (P_{Y,W}(z) - 2^{-s})^2 \cdot \mathbb{E}_{Y,W} \sum_z a(Y, W, z)^2} \\ &\leq \sqrt{2^{-\ell} \cdot 2^s q_0}, \end{aligned}$$

which proves $(*)$. □

The second lemma is a source-oblivious formulation of the standard connection between list decoding and hardcore functions, originating with Goldreich and Levin [25]. For the multi-bit distinguisher-to-predictor and reconstruction steps, see in particular [26, Section 7] and [30, Sections 5.1–5.2]. We include the proof (Section B.1) to make explicit the quantitative bounds and the formulation for arbitrary side information and an efficiently verifiable witness relation.

Lemma B.2 (Source-oblivious hardcore reduction). *Let (X, V) be any jointly distributed pair with $X \in \{0, 1\}^n$. Let $R(v, x')$ be an efficiently checkable relation such that $R(V, X) = 1$ always. Let*

$$C : \{0, 1\}^n \times \mathcal{W} \rightarrow \{0, 1\}^s$$

be a hash family whose induced code has the following list-decoding guarantee: an oracle with no erasures and agreement probability at least $2^{-s} + \rho$ with the codeword of x can be decoded, with probability at least $1/2$, with $O(n/\rho^2)$ oracle calls and list size, and $\text{poly}(n)/\rho^2$ additional time.

Suppose a distinguisher D has distinguishing advantage $\Delta > 0$ between

$$(V, W, C_W(X)) \quad \text{and} \quad (V, W, U_s),$$

where $W \leftarrow \mathcal{W}$ and $U_s \leftarrow \mathcal{U}_s$ are independent. Then there is an algorithm A that, given V , outputs x' satisfying $R(V, x') = 1$ with probability at least $\Delta/4$. Its running time is

$$O(n 2^{2s} \Delta^{-2})$$

invocations of D , plus $\text{poly}(n) 2^{2s} \Delta^{-2}$ overhead for list decoding and verification.

We can now prove Thm. 5.3.

Proof. Let

$$\text{Good} = \{x : \ell \leq r_g(x) \leq \ell + \tau n\}, \quad p = \Pr[X \in \text{Good}],$$

and let

$$X^* \leftarrow U_n \mid \text{Good}, \quad Y^* = g(X^*).$$

Note that $p \geq 1/2$ for sufficiently large n .

The event Good is a union of complete fibers of g . Hence, for every y in the support of Y^* ,

$$X^* \mid Y^* = y$$

is uniform on the full fiber $g^{-1}(y)$, and therefore

$$2^\ell \leq |g^{-1}(y)| \leq 2^{\ell + \tau n}. \tag{30}$$

(This is the only point at which fiber-closedness of Good is used.)

We now rerun the almost-regular analysis of [4, Theorem 5.2] on the conditioned source X^* . The two minor adaptations are spelled out below.

The first hash. Any inverter that succeeds on Y^* with probability q also succeeds on an ordinary challenge $g(U_n)$ with probability at least pq , simply by running it unchanged. Combined with $p \geq 1/2$, we have

$$q \leq \frac{2^{-bn}}{p} \leq 2^{1-bn}. \tag{31}$$

By the lower bound in (30), $X^* \mid Y^* = y$ is uniform over a set of size at least 2^ℓ for every y in the support of Y^* . If $s_1 = 0$, revealing $H_{W_1}(X^*)$ reveals nothing, so (31) already bounds the inversion probability. Otherwise, Lemma B.1, applied with $X = X^*$, $Y = Y^*$, and $s = s_1$, shows that every algorithm A of running time at most 2^{bn} , including polynomial overhead, satisfies

$$\Pr[A(Y^*, W_1, H_{W_1}(X^*)) \in g^{-1}(Y^*)] \leq \frac{2^{-bn}}{p} + \sqrt{\frac{2^{-bn}}{p} 2^{s_1 - \ell}}.$$

Since $p \geq 1/2$ and, whenever $s_1 > 0$, $s_1 - \ell \leq -3\tau n$, in all cases we obtain

$$\Pr[A(Y^*, W_1, H_{W_1}(X^*)) \in g^{-1}(Y^*)] \leq 2^{1-bn} + 2^{1/2-(b+3\tau)n/2}. \quad (32)$$

We will also need a statistical comparison for the final hybrid. If $s_1 = 0$, the following comparison is exact. Otherwise, the strong leftover hash lemma, applied separately in every fiber and averaged over Y^* , gives

$$\text{SD}((Y^*, W_1, H_{W_1}(X^*)), (Y^*, W_1, U_{s_1})) \leq \frac{1}{2} 2^{-(\ell-s_1)/2} \leq \frac{1}{2} 2^{-3\tau n/2}. \quad (33)$$

The hardcore block. Let

$$V = (Y^*, W_1, H_{W_1}(X^*)).$$

Suppose that a distinguisher of running time at most $2^{\tau n}$ has advantage at least $2^{-\tau n}$ between

$$(V, W_2, C_{W_2}(X^*)) \quad \text{and} \quad (V, W_2, U_{s_2}).$$

Apply Lemma B.2 with the relation

$$R(V, x') \iff g(x') = Y^*.$$

It yields an inverter for the distribution in (32) with success probability at least $2^{-\tau n}/4$ and running time

$$\text{poly}(n) \cdot 2^{\tau n} \cdot 2^{2s_2} \cdot 2^{2\tau n} \leq 2^{(15\tau+o(1))n} < 2^{bn},$$

where the last inequality follows from $\tau \leq b/24$. For sufficiently large n the upper bound in (32) is smaller than $2^{-\tau n}/4$. This contradiction proves

$$(V, W_2, C_{W_2}(X^*)) \approx_c (V, W_2, U_{s_2}) \quad (34)$$

for time $2^{\tau n}$ and advantage $2^{-\tau n}$.

Final extraction. By the upper bound in (30), for every y in the support of Y^* ,

$$\Pr[Y^* = y] = \frac{|g^{-1}(y)|}{p2^n} \leq \frac{2^{\ell+\tau n-n}}{p}.$$

Hence

$$H_\infty(Y^*) \geq n - \ell - \tau n + \log_2 p.$$

Let $R \leftarrow U_{s_1}$ be independent. Since

$$s_1 \geq \ell - 3\tau n - 1,$$

including the case $s_1 = 0$,

$$H_\infty(Y^*, R) \geq n - 4\tau n - 1 + \log_2 p. \quad (35)$$

The output length s_3 is smaller than the right-hand side by $\tau n - O(1)$. The strong leftover hash lemma therefore gives

$$(W_3, E_{W_3}(Y^*, R)) \approx_s (W_3, U_{s_3}) \quad (36)$$

with statistical distance $2^{-\Omega(\tau n)}$.

Combining (34), the statistical comparison (33), and (36), we conclude that

$$G(X, W_1, W_2, W_3) \mid X \in \text{Good}$$

is exponentially indistinguishable from uniform.

Finally, the actual generator samples $X \leftarrow U_n$ without conditioning. Since it evaluates g only once, replacing X by X^* changes the output distribution by statistical distance at most

$$\Pr[X \notin \text{Good}] \leq 2^{-cn}.$$

Thus the unconditional generator is exponentially secure.

The public hash keys appear in both the seed and the output, so the additive stretch is

$$\begin{aligned} s_2 + s_3 - n &\geq 6\tau n + (1 - 5\tau)n - n - O(1) \\ &= \tau n - O(1). \end{aligned}$$

This completes the proof of the theorem. □

B.1 Proof of Lemma B.2

Replace D by its complement if necessary so that the signed distinguishing gap is positive. For fixed (x, v) , define

$$\Delta(x, v) = \mathbb{E}_w D(v, w, C_w(x)) - \mathbb{E}_{w,z} D(v, w, z),$$

where the expectations include the internal randomness of D . Then

$$\mathbb{E}_{X,V} \Delta(X, V) = \Delta.$$

Since $\Delta(x, v) \leq 1$, the set of pairs satisfying

$$\Delta(x, v) \geq \Delta/2$$

has probability at least $\Delta/2$.

Fix such a pair (x, v) . We construct a randomized codeword oracle $\mathcal{O}_v$. On query w , sample independent uniform strings $z, z' \in \{0, 1\}^s$, run $D(v, w, z)$, output z if D accepts, and otherwise output z' . This oracle never erases. Let

$$p_{w,z} = \Pr[D(v, w, z) = 1], \quad \bar{p}_w = \mathbb{E}_z p_{w,z}.$$

For the target symbol $c = C_w(x)$, the oracle's agreement probability at coordinate w is

$$\begin{aligned} \Pr[\mathcal{O}_v(w) = c] &= 2^{-s} p_{w,c} + 2^{-s} (1 - \bar{p}_w) \\ &= 2^{-s} (1 + p_{w,c} - \bar{p}_w). \end{aligned}$$

Averaging over w gives

$$\Pr_w[\mathcal{O}_v(w) = C_w(x)] = 2^{-s} (1 + \Delta(x, v)) \geq 2^{-s} + 2^{-s} \Delta/2.$$

Thus the list decoder applies with

$$\rho = 2^{-s} \Delta / 2.$$

It produces a list containing x with probability at least $1/2$, using

$$O(n2^{2s} \Delta^{-2})$$

oracle calls. Check every list element using $R(v, \cdot)$ and output any valid one. Conditioned on a favorable pair (x, v) , this succeeds with probability at least $1/2$. Since favorable pairs have probability at least $\Delta/2$, the total success probability is at least $\Delta/4$. $\square$

C From typically almost-regular OWF to UOWHF: Proof of Theorem 6.2

Let

$$\text{Good} = \{x : \ell \leq r_g(x) \leq \ell + \alpha n\}, \quad p = \Pr[X \in \text{Good}], \quad \delta = 1 - p \leq 2^{-cn},$$

and let

$$X^* \leftarrow U_n \mid \text{Good}, \quad Y^* = g(X^*).$$

Note that $p \geq 1/2$ for sufficiently large n . The event Good is a union of complete fibers of g . Hence, for every y in the support of Y^* ,

$$X^* \mid Y^* = y$$

is uniform on the full fiber $g^{-1}(y)$, and therefore

$$2^\ell \leq |g^{-1}(y)| \leq 2^{\ell + \alpha n}. \quad (37)$$

(This is the only point at which fiber-closedness of Good is used.)

We analyze the two hashing stages on the conditioned source X^* . All reductions below have polynomial overhead, which is absorbed by the slack between the final adversary size $2^{\theta n}$ and the assumed one-wayness bound 2^{bn} .

The first hash. Any inverter of size at most $2^{bn}/2$ that succeeds on Y^* with probability q can be turned, with only polynomial overhead, into an inverter for $g(U_n)$ of size at most 2^{bn} and success probability at least pq . Hence

$$q \leq \varepsilon_0 := \frac{2^{-bn}}{p} \leq 2^{1-bn}. \quad (38)$$

Define

$$F_w(x) = (g(x), H_w(x)).$$

By the lower bound in (37), $X^* \mid Y^* = y$ is uniform over a set of size at least 2^ℓ for every y in the support of Y^* . Applying Lemma B.1 with $X = X^*$, $Y = Y^*$, and $s = s_1$, we obtain that every algorithm A of size at most $2^{bn}/2$ satisfies

$$\Pr[g(A(Y^*, W_1, H_{W_1}(X^*))) = Y^*] \leq \varepsilon_0 + \sqrt{\varepsilon_0 2^{s_1 - \ell}}.$$

Since $s_1 - \ell \leq 2\alpha n + 1$ and $p \geq 1/2$, the right-hand side is at most

$$2^{1-bn} + 2^{1-(b-2\alpha)n/2}.$$

In particular, an inverter for F_{W_1} is also such an algorithm. Removing the conditioning on Good therefore gives

$$\varepsilon_F := \delta + 2^{1-bn} + 2^{1-(b-2\alpha)n/2} \quad (39)$$

as an upper bound on the inversion probability of F_{W_1} on a uniform input.

We will also use that F_{W_1} is almost injective on a random input. Fix $x \in \text{Good}$. Any distinct x' satisfying

$$F_{W_1}(x') = F_{W_1}(x)$$

must lie in the same g -fiber as x and collide with x under H_{W_1} . By pairwise independence and a union bound,

$$\Pr_{W_1}[\exists x' \neq x : F_{W_1}(x') = F_{W_1}(x)] \leq |g^{-1}(g(x))| 2^{-s_1} \leq 2^{\ell+\alpha n-s_1} \leq 2^{-\alpha n}.$$

Averaging over $X \leftarrow U_n$ and charging the exceptional set,

$$\Pr_{X, W_1}[\exists x' \neq X : F_{W_1}(x') = F_{W_1}(X)] \leq \nu_F := \delta + 2^{-\alpha n}. \quad (40)$$

From a final-hash collision to inversion. Put $K = W_1$. We first analyze an adversary A given a uniform target $X \leftarrow U_n$ and independent keys (K, W_2) . Let

$$X' = A(X, K, W_2),$$

and let

$$q_{\neq} = \Pr[F_K(X') \neq F_K(X) \wedge E_{W_2}(F_K(X')) = E_{W_2}(F_K(X))].$$

Malformed answers are counted as failures. Among the remaining successful target collisions, those satisfying $F_K(X') = F_K(X)$ have probability at most ν_F by (40).

The inverter. We bound $q_{\neq}$ by reducing it to inversion of F . Define an inverter I as follows. Given an inversion challenge (k, y) for F , sample $x \leftarrow U_n$ and put $u = F_k(x)$. If $u = y$, abort. Otherwise, use the conditional key-samplability of E to sample a uniformly random key w subject to $E_w(u) = E_w(y)$. Run $A(x, k, w)$, obtaining x' , and output x' only if $F_k(x') = y$.

Analysis. We begin the analysis with the case where the image y is sampled uniformly at random. Let $Y_0 \leftarrow U_{m+s_1}$ independently of K . For distinct u, y , pairwise independence gives

$$\Pr_W[E_W(u) = E_W(y)] = 2^{-s_2}.$$

Hence, conditioned on this collision event, each key w satisfying $E_w(u) = E_w(y)$ receives a multiplicative weight 2^{s_2} relative to the uniform key distribution. On the other hand, the uniformly random challenge Y_0 must equal the particular value $F_K(X')$, contributing a factor $2^{-(m+s_1)}$. Therefore each execution counted by $q_{\neq}$ is reproduced by the inverter with total weight $2^{s_2-(m+s_1)}$, and so

$$\Pr[F_K(I(K, Y_0)) = Y_0] = 2^{s_2-(m+s_1)} q_{\neq}. \quad (41)$$

The abort when $u = y$ removes exactly the case $F_K(X') = F_K(X)$.

For every fixed k and every $y \in \text{Im}(F_k)$,

$$\Pr[F_k(X) = y] = \frac{|F_k^{-1}(y)|}{2^n} \geq 2^{-n}.$$

Consequently, the success probability of the same inverter on a uniform $(m + s_1)$ -bit string is at most $2^{n-(m+s_1)}$ times its success probability on an ordinary image challenge. By (39),

$$\Pr[F_K(I(K, Y_0)) = Y_0] \leq 2^{n-(m+s_1)} \varepsilon_F. \quad (42)$$

Combining (41) and (42) gives

$$q_{\neq} \leq 2^{n-s_2} \varepsilon_F.$$

Therefore the total random-target collision probability is at most

$$q \leq \nu_F + 2^{n-s_2} \varepsilon_F. \quad (43)$$

Parameters and the target-before-key convention. The assumptions imply $b \geq 16\alpha$ and $\alpha, c \geq 16\theta$. Hence

$$\varepsilon_F = O(2^{-16\theta n}), \quad \nu_F = O(2^{-16\theta n}).$$

Since $n - s_2 = \lceil \theta n \rceil \leq \theta n + 1$, (43) yields

$$q = O(2^{-15\theta n}).$$

All reductions applied to an adversary of size at most $2^{\theta n}$ have size below $2^{bn}/2$ for sufficiently large n . Thus, for example, $\gamma = \theta$ satisfies the claimed exponential security bound.

Finally, consider the standard experiment in which the target x_0 is chosen before the public key. The independent mask a turns this target into a uniform one. Formally, a random-target adversary given $x \leftarrow U_n$ first runs the original adversary's target-selection stage to obtain (x_0, st) , sets

$$a = x_0 \oplus x,$$

and invokes its second stage on (st, a, W_1, W_2) . If the latter returns x'_0 , output $x' = x'_0 \oplus a$. Since a is uniform and independent of x_0 , this exactly simulates the public mask in Construction 6.1, while preserving both distinctness and digest equality. Therefore the same collision bound holds in the standard target-before-key experiment. The key length is $O(n)$, and the shrinkage is $n - s_2 = \lceil \theta n \rceil$, as required. $\square$