Super-Linear Time-Memory Trade-Offs for Symmetric Encryption

Wei Dai[1]*, Stefano Tessaro[2], and Xihu Zhang[2]

[1] University of California, San Diego, La Jolla, USA
weidai@eng.ucsd.edu
[2] University of Washington, Seattle, USA
{tessaro,xihu}@cs.washington.edu

Abstract. We build symmetric encryption schemes from a pseudorandom function/permutation with domain size N which have very high security – in terms of the amount of messages q they can securely encrypt – assuming the adversary has $S < N$ bits of memory. We aim to *minimize* the number of calls k we make to the underlying primitive to achieve a certain q, or equivalently, to *maximize* the achievable q for a given k. We target in particular $q \gg N$, in contrast to recent works (Jaeger and Tessaro, EUROCRYPT '19; Dinur, EUROCRYPT '20) which aim to beat the birthday barrier with *one* call when $S < \sqrt{N}$.

Our first result gives new and explicit bounds for the Sample-then-Extract paradigm by Tessaro and Thiruvengadam (TCC '18). We show instantiations for which $q = \Omega\left((N/S)^k\right)$. If $S < N^{1-\alpha}$, Thiruvengadam and Tessaro's weaker bounds only guarantee $q > N$ when $k = \Omega(\log N)$. In contrast, here, we show this is true already for $k = \Theta(1/\alpha)$.

We also consider a scheme by Bellare, Goldreich and Krawczyk (CRYPTO '99) which evaluates the primitive on k independent random inputs, and masks the message with the XOR of the outputs. Here, we show $q = \Omega\left((N/S)^{k/2}\right)$, using new combinatorial bounds on the list-decodability of XOR codes which are of independent interest. We also study best-possible attacks against this construction.

1 Introduction

A number of very recent works [2,48,45,39,29,20,19,28] extend the concrete security treatment of provable security to account for the *memory complexity* of an adversary. For symmetric encryption, Jaeger and Tessaro [39] showed for example that randomized counter-mode encryption (CTR) is secure against attackers encrypting $q = \Theta(N/S)$ messages, where S is the memory complexity of the adversary and $N = 2^n$ is the domain size of the underlying PRF/PRP, which is assumed to be sufficiently secure. This is a *linear* time-memory trade-off – reducing S by a multiplicative factor $\varepsilon < 1$ allows us to increase by a factor $1/\varepsilon$ the tolerable data complexity of the attack.

* Work done in part while visiting the University of Washington.

The benefit of such a trade-off is that *if* $S < \sqrt{N}$, one can tolerate $q > \sqrt{N}$, which *is* beyond the so-called "birthday barrier." Building schemes with beyond-birthday security is a prime line of research in symmetric cryptography, but constructions are generally less efficient without imposing any memory restrictions on the adversary.

Our contributions: Super-linear trade-offs. The trade-off for CTR relies on a thin margin: For $N = 2^{128}$, we only improve upon memory-unbounded analyses if $S \ll 2^{64}$. While 2^{64} bits is a large amount of memory, it is not *unreasonably* large. One should therefore ask whether we can do better – either take advantage of a weaker memory limitation or be able to encrypt a much larger number of messages. More broadly, we want to paint a full picture of what security is attainable under a given memory restriction – complementing our understanding of the landscape *without* memory constraints.

More concretely, we consider constructions which make k calls to a given block cipher[1] with domain size N, and ask the following question:

> *If the adversary is bounded to $S < N$ bits of memory, what is the highest security we can achieve (in terms of allowable encryptions q) by a construction making k calls?*

Tessaro and Thiruvengadam [45] showed that one can achieve security for $q \gg N$ encrypted messages at the cost of $k = \Omega(\log N)$, whereas here we do much better by giving schemes that can do so already for $k = O(1)$: They can in particular encrypt up to $q = \Theta((N/S)^{c(k)})$ messages, for $c(k) > 1$. (This is what we refer to as a *super-linear* trade-off.) For one of our two constructions (in fact, the same construction as [45], but with a much better analysis), we get $c(k) = k - 1$ for messages of length n, and $c(k) = k$ for bit messages. These trade-offs appear best-possible (or close to best-possible), but proving optimality for now seems to be out of reach – we move first steps by studying attacks against one of our constructions.

These schemes can securely encrypt $q \gg N$ messages as long as $S < N$. It is important to appreciate that *without* the restriction, $q < N$ is an inherent barrier for current proof techniques (cf. [45] for a discussion).

On practice and theory. We stress that our approach is *foundational*. Even for $k \geqslant 2$, practitioners may find the resulting constructions not viable. Still, security beyond $q > N$ may be interesting in practice – we may want to implement a block cipher with smaller block length (e.g., $N = 2^{80}$) and then be able to still show security against $q = 2^{128}$ encryptions, as long as $S < 2^{80}$, which is a reasonable assumption.

We also stress that the question we consider here is natural in its own right, and is a cryptographic analogue and a scaled-up version of the line of works initiated by Raz [43], with a stronger focus on precise bounds and thus different techniques. (We discuss the connection further in Section 1.4 below.)

[1] Assumed to be a secure PRP/PRF.

1.1 Our Contributions

We start with a detailed overview of our contributions. (A technical overview is deferred to the next two sections.) Our constructions make k calls to a function $\mathsf{F}_K : \{0,1\}^n \to \{0,1\}^n$ keyed with a key K – this is generally obtained from a block cipher like AES (in which case, $n = 128$). We will use the shorthand $N = 2^n$. For the presentation of our results in this introduction, it is helpful to assume F_K behaves as a random function or a random permutation – this can be made formal via suitable PRF/PRP assumptions, and we discuss this at the end of this section in more detail.

The Sample-then-Extract Construction. The first part of this paper revisits the *Sample-then-Extract (StE)* construction of [45]. StE depends on a parameter $k \geqslant 1$ as well as a (strong) randomness extractor[2] $\mathsf{Ext} : (\{0,1\}^n)^k \times \{0,1\}^s \to \{0,1\}^\ell$. The encryption of a message $M \in \{0,1\}^\ell$ under key K is then

$$C = (R_1, \ldots, R_k, \mathsf{sd}, \mathsf{Ext}(\mathsf{F}_K(0 \,\|\, R_1) \,\|\, \cdots \,\|\, \mathsf{F}_K(k-1 \,\|\, R_k), \mathsf{sd}) \oplus M) \;, \tag{1}$$

where $\mathsf{sd} \in \{0,1\}^s$ and $R_1, \ldots, R_k \in \{0,1\}^{n-\log k}$ are chosen afresh upon each encryption. We also extend StE to encrypt arbitrary-length messages (which can have variable length), amortizing the cost of including $\mathsf{sd}, R_1, \ldots, R_k$, in the ciphertext. (For this introduction, however, we only deal with fixed-length messages for ease of exposition.)

Prior work only gives a sub-optimal analysis: For $k = \Theta(\log N) = \Theta(n)$, Tessaro and Thiruvengadam [45] show security against $q = N^{1.5}$ encryptions whenever $S = N^{1-\alpha}$ for a constant $\alpha > 0$. Here, we prove a much better bound. For example, for $\ell = n$, and a suitable choice of Ext, we show security up to

$$q = \Theta((N/S)^{k-1})$$

encryptions. This is improved to $q = \Theta((N/S)^k)$ for bit messages. Therefore, if $S < N^{1-\alpha}$, we can achieve security up to $q = N^{1.5}$ encryptions with $k = 1 + \frac{1.5}{\alpha}$, which is constant if α is constant.

The k-XOR Construction. Our second result considers a generalization of randomized counter-mode encryption, introduced by Bellare, Goldreich, and Krawczyk [7], which we refer to as the *k-XOR construction*. For even $k \geqslant 1$, to encrypt $M \in \{0,1\}^n$, we pick random $R_1, \ldots, R_k \in \{0,1\}^n$, and output

$$C = (R_1, \ldots, R_k, \mathsf{F}_K(R_1) \oplus \cdots \oplus \mathsf{F}_K(R_k) \oplus M) \;. \tag{2}$$

Alternatively, k-XOR can be viewed as an instance of StE with a seedless Ext. For this construction, we prove security up to $q = \Theta((N/S)^{k/2})$ encryptions. We note that in [7], a memory-independent bound of $q = \Theta(N/k)$ was proved for the case where $q \leqslant N$. The two results are complementary. The bound from [7] does not tell us anything for $q > N$, in contrast to our bound, but can beat (in

[2] Recall that this means that $(\mathsf{Ext}(X, \mathsf{sd}), \mathsf{sd})$ and (U, sd) are (statistically) indistinguishable for $\mathsf{sd} \xleftarrow{\$} \{0,1\}^s$, $U \xleftarrow{\$} \{0,1\}^\ell$, whenever X has sufficient min-entropy.

concrete terms) our bound for $q < N/k$. Different from our results on StE, our proof only works if we assume that F_K is a random *function*. We note however that this is consistent with the fact that even for the memory-unbounded setting, no bound based on a random permutation is known. We however discuss how to instantiate F_K from a PRP, and this will result in a construction similar to the above, just with a high number of calls to F.

It is also clear that we cannot expect to prove any better bound, unless we change the sampling of the indices $R_1, \ldots, R_k$. This is because after $q = N^{k/2}$ queries we will see, with very high probability, an encryption with $R_{2i-1} = R_{2i}$ for all $i = 1, \ldots, k/2$. This attack only requires $S = O(k \log N)$. However, it is not clear whether this attack extends to leverage larger values of S. Further discussion of attacks can be found in the full version.

Our proof relies on new *tight* combinatorial bounds on the list-decodability of XOR codes which are of independent interest and improve upon earlier works. Indeed, using existing best-possible bounds in our proof would result in a weaker bound with exponent $k/4$ (More details in the full version).

Reducing the ciphertext size. In the above constructions, the ciphertext size grows with k. An interesting question is whether we can avoid this – in the full version we do so for the case $S = \Omega(N)$. For this setting, our StE analysis gives $k = \Omega(n)$, and thus, the ciphertext has $\Omega(n^2)$ extra random bits in addition to the masked plaintext. In contrast, we present a variant of the StE construction where the number of extra bits in the ciphertext is reduced to $O(n)$. To this end, we use techniques from randomness extraction and randomness-efficient sampling to instantiate our construction.

Instantiating F_K. We instantiate F_K from a keyed function/permutation which we assume to be a pseudorandom function (PRF) or permutation (PRP). The catch is that if we aim for security against $q > N$ queries, we *need* F_K to be secure for adversaries that also run with time complexity larger than $t > q > N$.

This assumption is not unreasonable, as already discussed in [45] – one necessary condition is that the key is longer than $\log q$ bits to prevent a memory-less key-recovery distinguisher (e.g., one would use AES-256 instead of AES-128).[3] This is also easily seen to be sufficient in the ideal-cipher model, where PRP security *only* depends on the key length. Furthermore, our reductions give adversaries using memory $S < N$, and it is plausible that non-trivial attacks against block ciphers may use large amounts of memory. And finally, key-extension techniques [9,27,26,33] can give ciphers with security beyond N.

1.2 Our Techniques – Sample-then-extract

We discuss both constructions, StE and k-XOR, in separate sections, starting with the former.

Tighter hybrids. Our proof follows a paradigm (first introduced explicitly in [16], and then adapted in [39] to the memory-bounded setting) developing

[3] The best non-trivial attack against AES-256 uses time approximately 2^{254} [12].

hybrid-arguments in terms of Shannon-type metrics. This results in bounds of the form $\sqrt{q \cdot \varepsilon}$, whereas a classical hybrid arguments would give us bounds of the form $q\sqrt{\varepsilon}$. We do not know whether the square root *can* be removed – Dinur [19] shows how to do so in the Switching Lemma of [39], but it is unclear whether his techniques apply here.[4]

The core of our approach relies on understanding the distance from the uniform distribution for a sample with form

$$Y(\mathsf{F}) = (R_1, \ldots, R_k, \mathsf{sd}, \mathsf{Ext}(\mathsf{F}(0 \,\|\, R_1) \,\|\, \cdots \,\|\, \mathsf{F}(k-1 \,\|\, R_k), \mathsf{sd})) \ ,$$

for a randomly chosen function $\mathsf{F} : \{0,1\}^n \to \{0,1\}^n$, given additionally access to (arbitrary) S bits of leakage $\mathcal{L}(\mathsf{F})$. We will measure this distance in terms of KL divergence, by lower bounding the conditional Shannon entropy $\mathsf{H}(Y(\mathsf{F})|\mathcal{L}(\mathsf{F}))$. Giving a bound which is as large as possible will require the use of a number of tools in novel ways.

Decomposition lemma. For starters, we will crucially rely on the decomposition lemma of Göös et al. [32]: It shows that F_z – which is defined as F conditioned on $\mathcal{L}(\mathsf{F}) = z$ – is statistically γ-close to a convex combination of $(P, 1-\delta_z)$-dense random variable. A $(P, 1-\delta)$-dense random variable, in this context, is distributed over functions $\mathsf{F}' : \{0,1\}^n \to \{0,1\}^n$ and is such that there exists a set $\mathcal{P} \subseteq \{0,1\}^n$ of size P with the property that: (1) the outputs $\mathsf{F}'(x)$ are *fixed* for all $x \in \mathcal{P}$, whereas (2) for any subset $I \subseteq \{0,1\}^n \backslash \mathcal{P}$, the outputs $\{\mathsf{F}'(x)\}_{x \in I}$ have jointly min-entropy at least $|I| \cdot (1-\delta)n$. It is important to notice that there is a trade-off between γ, δ, and P, in that $\delta_z = (S_z + \log(1/\gamma))/(Pn)$, where $S_z = n2^n - \mathsf{H}_\infty(\mathsf{F}_z)$.

Extraction from varying amounts of min-entropy. Our analysis will choose the parameters δ and P carefully – the key point, however, is that when we replace F_z with a $(P, 1-\delta)$-dense function F', the total min-entropy of $\mathsf{F}'(0 \,\|\, R_1) \,\|\, \cdots \,\|\, \mathsf{F}'(k-1 \,\|\, R_k)$ grows with the number of probes R_i such that $(i \,\|\, R_i) \notin \mathcal{P}$, i.e., the set of "good" probes which land on an input for which the output is *not* fixed. To get some intuition, if one ignores the pre-pended probe index i, the number of good probes $g \in \{0, 1, \ldots, k\}$ would follow a binomial distribution with parameter $|\mathcal{P}|/N$, and overall min-entropy is $g \cdot (1-\delta)n$.

Therefore, the extractor is now applied to a random variable which has variable amount of min-entropy, which depends on g. Here, it is useful to use an extractor based on a 2-universal hash function: Indeed, the Leftover-Hash Lemma (LHL) [38] guarantees a very useful property, namely that while the extractor itself is *fixed*, the entropy of its output increases as the entropy of its input increases. Specifically, the entropy of the ℓ-bit output becomes $\ell - \min\{\ell, 2^{\ell+1-h}\}$ when the input has min-entropy $h \approx g(1-\delta)n$.

Our approach is dual to the smoothed min-entropy approach of Vadhan [47], which is used to build locally-computable extractors in a way that resembles

[4] This improvement is irrelevant as long as we only infer the resources needed for constant advantage, which is the standard angle on tightness in symmetric cryptography. However, as pointed out e.g. in [33], exact bounds also often matter.

ours. In our language, but with different techniques, he shows that with good probability, $g = \Theta(k)$, where $k = \Theta(\lambda)$. This does not work well for us (we care mostly about $k = O(1)$), and thus we take a more fine-grained approach geared towards understanding the behavior of g.

The advantage of Shannon entropy. It is crucial for the quality of the established trade-off to adopt a Shannon-entropy version of the LHL. The more common version bounds the statistical distance as $2^{(\ell+1-h)/2}$, and following this path would *only* give us a lower bound on q which is (roughly) the square root of what we prove. We note that a Shannon-theoretic version of the LHL was already proved by Bennet, Brassard, Crépeau, and Maurer [10], and the fact that a different distance metric can reduce the entropy loss is implicit in [4].[5]

Extra remarks. A few more remarks are in order. Our approach is similar, but also different from that of Coretti et al. [15,14]. They use the decomposition lemma in a similar way to transition to (what they refer to as) the *bit-fixing random oracle* (BF-RO), i.e., a model where F is fixed on P positions, and *completely random* on the remaining ones (as opposed to being just $(1-\delta)$-dense, as in our case). Using the BF-RO abstraction yields very suboptimal bounds. Their generic approach would incur an additive factor of $(S + \log(1/\gamma))k/P$, which is too large.

1.3 Our techniques - k-XOR

Our approach for StE given above does not yield usable results for k-XOR – namely, any choice of δ prevents us from proving that $\mathsf{F}_z(0 \,\|\, R_1) \oplus \cdots \oplus \mathsf{F}_z(k-1 \,\|\, R_k)$ is very close to uniform, even if none of the probes lands in $\mathcal{P}$. A unifying treatment of both constructions appears to require finding a strengthening of the decomposition lemma. Instead, we follow a different path.

Predicting XORs. The core of our analysis bounds the ability of predicting $\mathsf{F}(R_1) \oplus \cdots \oplus \mathsf{F}(R_k)$ for a random function $\mathsf{F} : \{0,1\}^n \to \{0,1\}$, given (arbitrary) S bits of leakage on F. We aim to upper bound the advantage $\Delta(N, S, k)$ which measures how much beyond probability $\frac{1}{2}$ an adversary can guess the XOR given the leakage and $R_1, \ldots, R_k$. The focus is on *single-bit* outputs – a bound for the multi-bit case will follow from a hybrid argument. Although this problem has been studied [22,46,35,37,17], both in the contexts of locally-computable extractors for the bounded-storage model and of randomness extraction, none of these techniques gives bounds which are tight enough for us. (We elaborate on this below.) Here, we shall prove that

$$\Delta(N, S, k) = O((S/N)^{k/2}) \,.$$

The coding connection. Our solution leverages a connection with the list-decoding of the *k-fold XOR code* (or k-XOR code, for short): This encodes F (which we think now as an N-bit string $F \in \{0,1\}^N$) as an N^k-dimensional

[5] The benefits of reducing entropy loss by targeting Shannon-like metrics were also very recently studied by Agrawal [1] in a different context.

bit-vector $\mathsf{k\text{-}XOR}(F) \in \{0,1\}^{N^k}$ such that its component $(R_1, \ldots, R_k) \in [N]^k$ takes value $F(R_1) \oplus \cdots \oplus F(R_k)$. At the same time, a (deterministic) adversary $\mathcal{A}$ which on input $R_1, \ldots, R_k$ and the leakage $Z = L(F)$ attempts to predict $F(R_1) \oplus \cdots \oplus F(R_k)$ can be thought of as family of 2^S "noisy strings" $\{C_Z = \mathcal{A}(\cdot, Z)\}_{Z \in \{0,1\}^S}$.

Prior works (such as [17]) focused (directly or indirectly) on *approximate* list-decoding, as they give *reductions*, transforming $\mathcal{A}$ *and* L into some predictor for F, under some slightly larger leakage. (How much larger the leakage is depends on the approximate list size.) Here, instead, we follow a combinatorial blueprint inspired by [8,6], albeit very different in its execution. Concretely, we introduce a parameter $\varepsilon > 0$ (to be set to a more concrete value later), and for all $Z \in \{0,1\}^S$, let $\mathcal{B}_Z$ be the Hamming Ball of radius $(1/2 - \varepsilon)N^k$ around C_Z. Now, when picking $F \xleftarrow{\$} \{0,1\}^N$, exactly one of two cases can arise:

- **(i)** $\mathsf{k\text{-}XOR}(F) \in \mathcal{B}_Z$ for some $Z \in \{0,1\}^S$, in which case the overlap between C_Z and $\mathsf{k\text{-}XOR}(F)$ is potentially very high.
- **(ii)** $F \notin \bigcup_Z \mathcal{B}_Z$, in which case $\mathcal{A}$ will be able to predict $F(R_1) \oplus \cdots \oplus F(R_k)$ with probability at most $1/2 + \varepsilon$ over the random choice of $R_1, \ldots, R_k$ - *no matter* how $L(F)$ is defined!

Now, let L^k_ε be an upper bound on the number of codewords $\mathsf{k\text{-}XOR}(F)$ within any of the $\mathcal{B}_Z$. Then,

$$\Delta(N, S, k) \leqslant \varepsilon + 2^S \cdot L^k_\varepsilon / 2^N \ . \tag{3}$$

Tight bounds on list-decoding size. What remains to be done here is to find a bound on L^k_ε – we are not aware of any tight bounds in the literature, and we give such bounds here.

Our approach (and its challenges) are illustrated best in the case $k = 1$. Specifically, define random variables $T_1, \ldots, T_N$, where, for all $R \in [N]$, $T_R = 1$ if $C_Z(R) = F(R)$ and $T_R = 0$ else. When we pick F at random, the T_i's are independent, and a Chernoff bound tells us that

$$\mathsf{Pr}\left[\sum_{R=1}^{N} T_R \geqslant \left(\frac{1}{2} + \varepsilon\right) N\right] \leqslant 2^{-\Omega(\varepsilon^2 N)} \ ,$$

which in turn implies $L^1_\varepsilon \leqslant 2^{N(1-\varepsilon^2)}$. Therefore, setting ε to be of order slightly larger than $\sqrt{S/N}$ gives us the right bound.

Our proof for $k > 1$ will follow a similar blueprint, except that this will require us to prove a (much harder!) concentration bound on a sum of N^k variables which are highly dependent. We will prove such concentration using the method of moments. The final bound will be of the form $L^k_\varepsilon \leqslant 2^{N(1-\varepsilon^{2/k})}$.

Relationship to past works. We are not aware of any prior work addressing the question of proving tight bounds for the XOR code *directly*, but prior techniques can non-trivially be combined to obtain non-trivial bounds. The best-possible bound we could derive is $(S/N)^{k/4}$. This can be obtained by combining the approach of De and Trevisan [17] with the *combinatorial* approximate

list-decoding bounds of [37]. Alternatively, one could use the approximate list decoding bounds from [11]. The resulting indistinguishability bound is harder to evaluate, but it is inferior for small values of S (roughly, $S < N^{2/3}$). Further details are in the full version.

Optimality. We discuss attacks against k-XOR in the full version. In particular, one can easily see that if we want the bound to hold *for all* values of S, then it cannot be improved, as it is tight for small $S = O(k \log N)$. For a broader range of values of S, we give an attack which succeeds with $q = \Theta((N^k/S^{k-1})$ messages and for $k = 2$ we provide an attack that succeeds with $q = \Theta((N/S)^2)$ – it is a good question whether our bound can be improved for larger values of S, or in the case where the $R_1, \ldots, R_k$ are *distinct*. (This would preclude our small-memory attack.)

Our general attack that works for any S and k, stores all linear equations that have all variables fall in $x_1, \ldots, x_S$ and checks consistency. It is expected that a linear dependent equation would appear within $q = O(N^k/S^{k-1})$ queries. Our next attack addresses the case where $k = 2$. By modeling each variable as a vertex and representing each equation as an edge in the graph, the attack exploits the tree structure formed by linear independent equations and succeeds within $q = O((N/S)^2)$ queries. However, for $k \geqslant 3$, similar analysis no longer applies as the hypergraph structure is hard to analyze.

1.4 Further Related work

Space-time trade-offs for learning problems. A related line of works is that initiated by Raz [43] on space-time trade-offs for learning problems, which has by now seen several follow-ups [44,40,5,25,24]. In particular, Raz proposes a scheme encrypting each bit m_i as $(a_i, \langle a_i, s\rangle + m_i)$ where $s \xleftarrow{\$} \{0,1\}^n$ is a secret key, and $a_i \xleftarrow{\$} \{0,1\}^n$ is freshly sampled for each bit. This scheme allows to encrypt 2^n bits as long as the adversary's memory is at most n^2/c bits, for some (small) constant $c > 1$. We *can* scale up this setting to ours, by thinking of s as the exponentially large table of a random function, but the resulting scheme would also incur exponential complexity. Some follow-up works consider the cases where the a_i's are *sparse* [5,25], but they only study the problem of *recovering* s, and it does not seem possible to obtain (sufficiently sharp) indistinguishability bounds from these results.

Closest to our work on k-XOR is a recent concurrent paper [24] by Garg, Kothari and Raz, which studies the streaming indistinguishability of Goldreich's PRG [30] against memory bounded adversaries. Their target are bounds for arbitrary predicates for Goldreich's PRG, and they prove indistinguishability for up to $q = \Theta\left((N/S)^{k/9}\right)$ output bits when the predicate is k-XOR. The setting of the analysis is almost identical to ours, with the difference being that we think of the PRG seed as being an exponentially large random table. Thus our

techniques also yield a tighter bound in their setting for this special case,[6] and we believe they should also yield improved bounds for more general predicates.

On the flip side, it is an exciting open question whether the branching-program framework underlying all of these works can be adapted to obtain bounds as sharp as ours in the indistinguishability setting.

The Bounded-Storage Model. In both cases, our proofs consider the intermediate setting where S bits of leakage $Z = \mathcal{L}(F)$ are given about F, and we want to show that the output of some locally computable function $g(F, R)$ is random enough given Z, where R is potentially public randomness. This is exactly what is considered in the *Bounded Storage Model* (BSM) [42,3,47,23,17] and in the *bounded-retrieval model* (BRM) [21,18]. Indeed, our StE construction can be traced back to the approach of locally-computable extractors [47], and the k-XOR construction resembles the constructions of [42,3,23]. A substantial difference, however, is that we are inherently concerned about the small-probe setting (i.e., $k = O(1)$) and the case where $S = N^{1-\alpha}$, whereas generally the BSM considers $S = O(N)$ and a *linear* number of probes. We also take a more concrete approach towards showing as-tight-as-possible bounds for a given target k. It would be beneficial to address whether our techniques can be used to improve existing BSM/BRM schemes.

Another difference is that our bounds are typically multiplied by the number of encryption queries. This can be done non-trivially, for example, by using Shannon entropy as a measure of randomness, and relying on the reduced entropy loss for extraction with respect to Shannon entropy, as we do for StE.

2 Definitions

Let $\mathbb{N} = \{0, 1, 2, \dots\}$. For $N \in \mathbb{N}$ let $[N] = \{1, 2, \dots, N\}$. If A and B are finite sets, then $\mathsf{Fcs}(A, B)$ denotes the set of all functions $F : A \to B$ and $\mathsf{Perm}(A)$ denotes the set of all permutations on the set A. The set of size k subsets of A is $\binom{A}{k}$. Picking an element uniformly at random from A and assigning it to s is denoted by $s \xleftarrow{\$} A$. The set of finite vectors with entries in A is $(A)^*$ or A^*. Thus $\{0,1\}^*$ is the set of finite length strings.

If $M \in \{0,1\}^*$ is a string, then $|M|$ denotes its bit length. If $m \in \mathbb{N}$ and $M \in (\{0,1\}^m)^*$, then $|M|_m = |M|/m$ denote the block length of M and M_i denote the i-th m-bit block of M. When using the latter notation, m will be clear from context. The *Hamming weight* $\mathsf{hw}(x)$ of $x \in \{0,1\}^n$ is defined as $\mathsf{hw}(x) = |\{i \in [n] \mid x_i \neq 0\}|$. The *Hamming ball* of radius r around $z \in \{0,1\}^n$ is defined as $\mathcal{B}(z; r) = \{x \in \{0,1\}^n \mid \mathsf{hw}(x \oplus z) \leqslant r\}$.

We say that a random variable X is a *convex combination* of random variables $X_1, ..., X_t$ (with the same range as X) if there exists $\alpha_1, ..., \alpha_t \geqslant 0$ such that $\sum_{i=1}^{t} \alpha_i = 1$ and for any x in the range of X, it holds that $\mathsf{Pr}[X = x] = \sum_{i=1}^{t} \alpha_i \mathsf{Pr}[X_i = x]$.

[6] There is a small formal difference, in that our analysis of k-XOR evaluates the given function on random indices, whereas in [24] these indices are distinct.

GAMES. Our cryptographic reductions will use pseudocode games (inspired by the code-based framework of [9]). See Fig. 1 for some example games. We let $\mathsf{Pr}[\mathsf{G}]$ denote the probability that game G outputs `true`. It is to be understood that the model underlying this pseudocode is the formalism we now describe.

COMPUTATIONAL MODEL. Our algorithms are randomized when not specified otherwise. If $\mathcal{A}$ is an algorithm, then $y \leftarrow \mathcal{A}^{\mathrm{O}_1,\mathrm{O}_2,\cdots}(x_1,\ldots;r)$ denotes running $\mathcal{A}$ on inputs $x_1,\ldots$ and coins r with access to oracles $\mathrm{O}_1,\mathrm{O}_2,\ldots$ to produce output y. The notation $y \xleftarrow{\$} \mathcal{A}^{\mathrm{O}_1,\mathrm{O}_2,\cdots}(x_1,\ldots)$ denotes picking r at random then running $y \leftarrow \mathcal{A}^{\mathrm{O}_1,\mathrm{O}_2,\cdots}(x_1,\ldots;r)$. The set of all possible outputs of $\mathcal{A}$ when run with inputs $x_1,\ldots$ is $[\mathcal{A}(x_1,\ldots)]$. Adversaries and distinguishers are algorithms. The notation $y \leftarrow \mathrm{O}(x_1,\ldots)$ is used for calling oracle O with inputs $x_1,\ldots$ and assigning its output to y (even if the value assigned to y is not deterministically chosen).

We say that an algorithm (or adversary) $\mathcal{A}$ runs in time t if its description size and running time are at most t. We say that adversary $\mathcal{A}$ is *S-bounded* if it uses at most S bits of memory during its execution, for any possible oracle it is given access to and any possible input.

INFORMATION THEORY. For a random variable X with probability distribution $P(x) = \mathsf{Pr}[X = x]$, the *Shannon entropy* $\mathsf{H}(X)$ and *collision entropy* $\mathsf{H}_2(X)$ are defined as $\mathsf{H}(X) = -\sum_x P(x)\log P(x)$ and $\mathsf{H}_2(X) = -\log\left(\sum_x P(x)^2\right)$. The min-entropy of X is $\mathsf{H}_\infty(X) = -\log\max_x P(x)$. For two random variables X, Y with joint distribution $Q(x,y) = \mathsf{Pr}[X = x, Y = y]$, the *conditional Shannon entropy* and *conditional min-entropy* are defined by $\mathsf{H}(Y|X) = \sum_{x,y} Q(x,y)\log\frac{Q(x)}{Q(x,y)}$ and $\mathsf{H}_\infty(Y|X) = -\log\sum_x \max_y Q(x,y)$, where $Q(x) = \sum_y Q(x,y)$ is the marginal distribution of X.

2.1 Streaming indistinguishability

We review the streaming indistinguishability framework of Jaeger and Tessaro [39], which considers a setting where a sequence, $\mathbf{X}$, of random variables

$$X_1, X_2, \ldots, X_q$$

with range $[N]$ is given, one by one, to a (memory-bounded) distinguisher $\mathcal{A}$. The distinguisher will need to tell apart this setting from another one, where it is given $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_q)$ instead.

THE STREAMING MODEL. More formally, in the i-th step (for $i \in [q]$), the distinguisher $\mathcal{A}$ has a state σ_{i-1} and stage number i. Then it receives $V_i \in \{X_i, Y_i\}$ based on which it updates its state to σ_i. We denote by $\sigma_i(\mathcal{A}(\mathbf{X}))$ and $\sigma_i(\mathcal{A}(\mathbf{Y}))$ the state after receiving X_i and Y_i when running $\mathcal{A}$ on streams $\mathbf{X}$ and $\mathbf{Y}$, respectively. We say here that $\mathcal{A}$ is S-bounded if all states have bit-length at most S.[7] We also assume that $\sigma_q \in \{0,1\}$, and think of σ_q as the output of $\mathcal{A}$. We

[7] Note, quite crucially, that this is different from the definition of S-bounded algorithms, in that we relax our notion of space-boundedness to only consider the states between stages. This is sufficient for our applications, although the model can be restricted.

Game $\mathsf{G}^{\mathsf{fn}}_{\mathsf{F}}(\mathcal{A})$	Game $\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},b}(\mathcal{A})$
$K \xleftarrow{\$} \mathsf{F.Ks}$	$K \xleftarrow{\$} \mathsf{SE.Ks}$
$b \xleftarrow{\$} \mathcal{A}^{\text{FN}}$	$b' \xleftarrow{\$} \mathcal{A}^{\text{ENC}}$
Return $b = 1$	Return $b' = 1$
FN(X)	ENC(M)
$Y \leftarrow \mathsf{F}(K, X)$	$C_1 \leftarrow \mathsf{SE.Enc}(K, M)$
Return Y	$C_0 \xleftarrow{\$} \{0,1\}^{\lvert M\rvert + \mathsf{SE.xl}}$
	Return C_b

Fig. 1. Security games for PRF/PRP security of a family of functions (Left) and INDR security of an encryption scheme (Right).

---

define the following streaming-distinguishing advantage

$$\mathsf{Adv}^{\mathsf{dist}}_{\mathbf{X},\mathbf{Y}}(\mathcal{A}) = \mathsf{Pr}\left[\mathcal{A}(\mathbf{X}) \Rightarrow 1\right] - \mathsf{Pr}\left[\mathcal{A}(\mathbf{Y}) \Rightarrow 1\right] .$$

We shall use the following lemma by [39].

Lemma 1. *Let* $\mathbf{X} = (X_1, \ldots, X_q)$ *be independent and uniformly distributed over* $[N]$ *and let* $\mathbf{Y} = (Y_1, \ldots, Y_q)$ *be distributed over the same support as* $\mathbf{X}$*. Then,*

$$\mathsf{Adv}^{\mathsf{dist}}_{\mathbf{X},\mathbf{Y}}(\mathcal{A}) \leqslant \frac{1}{\sqrt{2}} \sqrt{q \log N - \sum_{i=1}^{q} \mathsf{H}(Y_i \mid \sigma_{i-1}(\mathcal{A}(\mathbf{Y})))} .$$

2.2 Cryptographic preliminaries

FAMILY OF FUNCTIONS. A function family F is a function of the form $\mathsf{F} : \mathsf{F.Ks} \times \mathsf{F.Dom} \to \mathsf{F.Rng}$. It is understood that there is some algorithm that samples from the set $\mathsf{F.Ks}$, and that fixing $K \in \mathsf{F.Ks}$, there is some algorithm that computes the function $\mathsf{F}_K(\cdot) = \mathsf{F}(K, \cdot)$. For our purposes, it suffices to restrict to function families where $\mathsf{F.Dom} = \{0,1\}^n$ and $\mathsf{F.Rng} = \{0,1\}^m$ for some n and m.

A blockcipher is a family of functions F for which $\mathsf{F.Dom} = \mathsf{F.Rng}$ and for all $K \in \mathsf{F.Ks}$ the function $\mathsf{F}(K, \cdot)$ is a permutation.

We let $\mathsf{RF}_{n,m} : \mathsf{Fcs}(\{0,1\}^n, \{0,1\}^m) \times \{0,1\}^n \to \{0,1\}^m$ be the function family of all functions mapping n-bits to m-bits, i.e. for any $F \in \mathsf{Fcs}(\{0,1\}^n, \{0,1\}^m)$ and $x \in \{0,1\}^n$, we define $\mathsf{RF}_{n,m}(F, x) = F(x)$. We let $\mathsf{RP}_n : \mathsf{Perm}(\{0,1\}^n) \times \{0,1\}^n \to \{0,1\}^n$ be the function family of all permutations on n bits. It is defined so that for any $P \in \mathsf{Perm}(\{0,1\}^n)$ and $x \in \{0,1\}^n$, $\mathsf{RP}_n(P, x) = P(x)$.

PSEUDORANDOMNESS SECURITY. For security we will consider both pseudorandom function (PRF) and pseudorandom permutation (PRP) security.

Let F be a function family with $\mathsf{F.Dom} = \{0,1\}^n$ and $\mathsf{F.Rng} = \{0,1\}^m$. PRF security asks F to be indistinguishable from $\mathsf{RF}_{n,m}$. More formally, consider the function evaluation game $\mathsf{G}^{\mathsf{fn}}_{\mathsf{F}}(\mathcal{A})$, in which adversary simply gets access to an

oracle evaluating F_K for a random and fixed key K. The PRF advantage of $\mathcal{A}$ against F is defined to be

$$\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}) = \mathsf{Pr}[\mathsf{G}^{\mathsf{fn}}_{\mathsf{F}}(\mathcal{A})] - \mathsf{Pr}[\mathsf{G}^{\mathsf{fn}}_{\mathsf{RF}_{n,m}}(\mathcal{A})] \ .$$

Similarly, PRP security of a blockcipher F with $\mathsf{F.Dom} = \{0,1\}^n$ is defined to be

$$\mathsf{Adv}^{\mathsf{prp}}_{\mathsf{F}}(\mathcal{A}) = \mathsf{Pr}[\mathsf{G}^{\mathsf{fn}}_{\mathsf{F}}(\mathcal{A})] - \mathsf{Pr}[\mathsf{G}^{\mathsf{fn}}_{\mathsf{RP}_n}(\mathcal{A})] \ .$$

SYMMETRIC ENCRYPTION. A symmetric encryption scheme SE specifies key space $\mathsf{SE.Ks}$, and algorithms $\mathsf{SE.Enc}$, and $\mathsf{SE.Dec}$ (where the last of these is deterministic) as well as set $\mathsf{SE.M}$. Encryption algorithm $\mathsf{SE.Enc}$ takes as input key $K \in \mathsf{SE.Ks}$ and message $M \in \mathsf{SE.M}$ to output a ciphertext C. We assume there exists a constant expansion length $\mathsf{SE.xl} \in \mathbb{N}$ such that $|C| = |M| + \mathsf{SE.xl}$. Decryption algorithm $\mathsf{SE.Dec}$ takes as input ciphertext C to output $M \in \mathsf{SE.M} \cup \{\perp\}$. We write $K \xleftarrow{\$} \mathsf{SE.Ks}$, $C \xleftarrow{\$} \mathsf{SE.Enc}(K, M)$, and $M \leftarrow \mathsf{SE.Dec}(C)$.

Correctness requires for all $K \in \mathsf{SE.Ks}$ and all sequences of messages $\boldsymbol{M} \in (\mathsf{SE.M})^*$ that $\mathsf{Pr}[\forall i : \boldsymbol{M}_i = \boldsymbol{M}'_i] = 1$ where the probability is over the coins of encryption in the operations $\boldsymbol{C}_i \xleftarrow{\$} \mathsf{SE.Enc}(K, \boldsymbol{M}_i)$ and $\boldsymbol{M}'_i \leftarrow \mathsf{SE.Dec}(K, \boldsymbol{C}_i)$ for $i = 1, \ldots, |\boldsymbol{M}|$.

For security we will require the output of encryption to look like a random string. Consider the game $\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},b}(\mathcal{A})$ shown on the right side of Figure 1. It is parameterized by a symmetric encryption scheme SE, adversary $\mathcal{A}$, and bit $b \in \{0,1\}$. The adversary is given access to an oracle ENC which, on input a message M, returns either the encryption of that message or a random string of the appropriate length according to the secret bit b. The advantage of $\mathcal{A}$ against SE is defined by $\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{SE}}(\mathcal{A}) = \mathsf{Pr}[\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},1}(\mathcal{A})] - \mathsf{Pr}[\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},0}(\mathcal{A})]$.

3 Sample-Then-Extract

The $\mathsf{StE} = \mathsf{StE}[\mathsf{F}, k, \mathsf{Ext}]$ scheme is defined in Figure 2: It was originally proposed by Tessaro and Thiruvengadam [45], and it is based on ideas from the context of locally-computable extractors [47]. The scheme is extended here to encrypt multiple blocks of message with the same randomness $R_1 \ldots, R_k$, and the same extractor seed sd. The scheme $\mathsf{StE}[\mathsf{F}, k, \mathsf{Ext}]$ uses a keyed function family F which maps $\{0,1\}^n$ to $\{0,1\}^n$, as well as an extractor $\mathsf{Ext} : \{0,1\}^{kn} \times \{0,1\}^s \to \{0,1\}^\ell$.

Below, we instantiate the extractor Ext with 2-universal hash function [13]. We recall that $\mathsf{h} : \{0,1\}^w \times \{0,1\}^s \to \{0,1\}^\ell$ is *2-universal* if for all distinct $x, y \in \{0,1\}^w$, it holds that $\mathsf{Pr}[\mathsf{sd} \xleftarrow{\$} \{0,1\}^s : \mathsf{h}(x, \mathsf{sd}) = \mathsf{h}(y, \mathsf{sd})] = 2^{-\ell}$. For conciseness, we often write $\mathsf{h}_{\mathsf{sd}}(x) = \mathsf{h}(x, \mathsf{sd})$. If $\ell \leqslant s$, a construction with $w = s$ interprets both the input x and the seed sd as elements of the extension field $\mathbb{F}_{2^w}$, and $\mathsf{h}(x, \mathsf{sd})$ consists of the first ℓ bits of the product of x and sd.

A SMALL-CIPHERTEXT VERSION OF StE. We also study a version of StE which produces small ciphertexts, using techniques from randomness efficient sampling. The proof resembles that for StE given below, and the details are deferred to the full version due to limited space.

Scheme $\mathsf{StE}[\mathsf{F}, k, \mathsf{Ext}]$

Procedure $\mathsf{Enc}(K, M)$

$B \leftarrow |M|_\ell$

$M_1, \ldots, M_B \leftarrow M$; $\mathsf{sd} \xleftarrow{\$} \{0,1\}^s$

$\mathbf{R} = (R_1, ..., R_k) \xleftarrow{\$} \left(\{0,1\}^{n-\lceil \log k \rceil}\right)^k$

For $i \in [B]$ do

  For $j \in [k]$ do

    $V_{i,j} \leftarrow \mathsf{F}(K, (j-1) \| (R_j + i - 1))$

For $i \in [B]$ do

  $C_i \leftarrow M_i \oplus \mathsf{Ext}(V_{i,1} \| ... \| V_{i,k}, \mathsf{sd})$

Return $(\mathsf{sd}, \mathbf{R}, C_1, \ldots, C_B)$

Procedure $\mathsf{Dec}(K, C)$

$(\mathsf{sd}, \mathbf{R}, C_1, \ldots, C_B) \leftarrow C$

For $i \in [B]$ do

  For $j \in [k]$ do

    $V_{i,j} \leftarrow \mathsf{F}(K, (j-1) \| (R_j + i - 1))$

For $i \in [B]$ do

  $M_i \leftarrow C_i \oplus \mathsf{Ext}(V_{i,1} \| \cdots \| V_{i,k}, \mathsf{sd})$

Return $M_1 \| \cdots \| M_B$

Fig. 2. The sample-then-extract encryption scheme $\mathsf{SE} = \mathsf{StE}[\mathsf{F}, k, \mathsf{Ext}]$, with $\mathsf{F.Dom} = \{0,1\}^n$. All additions and subtractions are done under modulus $2^{n-\lceil \log k \rceil}$. The key space and message space of SE are $\mathsf{SE.Ks} = \mathsf{F.Ks}$ and $\mathsf{SE.M} = (\{0,1\}^\ell)^+$.

3.1 Security of StE

The security of StE scheme is captured by the following theorem. We first consider the case where F is a PRF – which we prove below first. We will state a very similar theorem for the PRP case below.[8]

The proof of the main theorem is deferred to Section 3.2.

Theorem 1 (Security of StE). *Let $N = 2^n$, let $\mathsf{F} : \mathsf{F.Ks} \times \{0,1\}^n \to \{0,1\}^n$ be a keyed function family. Let Ext be a 2-universal hash function $\mathsf{h} : \{0,1\}^{kn} \times \{0,1\}^{kn} \to \{0,1\}^\ell$. For any S-bounded q-query adversary $\mathcal{A}_{\mathsf{indr}}$, where each query consists of messages of at most B ℓ-bit blocks such that $B \leqslant N/k$, there exists an $(S + B\ell)$-bounded PRF adversary $\mathcal{A}_{\mathsf{prf}}$ (with similar time complexity as $\mathcal{A}_{\mathsf{indr}}$) that issues at most qkB queries to the oracle, such that*

$$\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{StE}[\mathsf{F},k,\mathsf{h}]}(\mathcal{A}_{\mathsf{indr}}) \leqslant \mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prf}}) + \sqrt{\frac{1}{2} qB\varepsilon}\ ,$$

where

$$\varepsilon = \frac{\ell}{N^k} + \sum_{t=0}^{k} \binom{k}{t} \left(\frac{(2S + 2kn)B}{N}\right)^t \cdot \min\{\ell, 2^{\ell+1} \cdot (2/N)^{k-t}\}\ .$$

Instantiations and interpretations. We discuss instantiations of the above theorem for specific parameter regimes. We consider two choices of ℓ, which result in different bounds. In fact, a subtle aspect of the bound is the appearance of a

[8] The PRP assumption leads to more straightforward instantiations via a block cipher. The PRF instantiation is trickier, as we need PRFs that are highly secure – these can be instantiated with a much higher cost from a good PRP.

min: Depending on the choice of ℓ (relative to N), we will have different t^* such that $2^{\ell+1} \cdot (2/N)^{k-t} > \ell$ for all $t < t^*$, and the value t^* affects the bound.

We give two corollaries. The first one dispenses with any fine-tuning, and just upper bounds the min with $2^{\ell+1} \cdot (2/N)^{k-t}$. This bound however is enough to give us a strong trade-off of $q = \Omega(N^k/S^k)$ for $\ell = O(1)$. However, for another common target, $\ell = n$, this would give us $q = \Omega(N^{k-1}/S^k)$. Our second corollary will show how the setting t^* in that case will lead to a stronger lower bound of $q = \Omega(N^{k-1}/S^{k-1})$. (In both cases, we are stating this for $B = 1$.)

Corollary 1. *With the same setup as Theorem 1, we have*

$$\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{StE}[\mathsf{F},k,\mathsf{h}]}(\mathcal{A}_{\mathsf{indr}}) \leqslant \mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prf}}) + \sqrt{2^{\ell} qB \left(\frac{(2S+2kn)B+3}{N}\right)^{k}} .$$

Corollary 2. *With the same setup as Theorem 1, in addition to $n = \ell$, $n \geqslant 4$, and $k \geqslant 2$, we have*

$$\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{StE}[\mathsf{F},k,\mathsf{h}]}(\mathcal{A}_{\mathsf{indr}}) \leqslant \mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prf}}) + \sqrt{2qBk \left(\frac{(2S+2kn)B+4n}{N}\right)^{k-1}} .$$

We defer the proof of both corollaries to the full version.

We further provides an analysis over parameters of practical interests. Concretely, if we instantiate F by a PRF that maps 128-bit to 128-bit, that is, $N = 2^{128}$, and we let the block size $\ell = 128$ bit. Then for any adversary that uses at most $S = 2^{80}$ bit of memory and encrypts at most 1GB message per query (i.e. $B = 2^{33-7} = 2^{26}$), by following the coarse analysis of Corollary 1 and letting $k = 15$, our scheme can tolerate roughly $q = 2^{(128-80-26-1)\cdot 15-128-26} = 2^{161}$ queries. However, we do not need such a large k to achieve $q > N$. Notice that $\ell = n = 128$, we can use Corollary 2 to improve the analysis. Then by setting $k = 9$, we have $q = 2^{(128-80-26-1)\cdot(k-1)-26-1} = 2^{21\cdot 8-27} = 2^{141}$ queries encrypting 1GB message. Note that similar analysis can be obtained when adapting the following PRP instantiation.

PRP INSTANTIATION. The security of StE instantiated by a PRP is captured by the following theorem. Since the StE-PRP security proof is similar to StE-PRF proof (the latter is slightly easier to present), we provide a proof sketch for the PRP case in the full version, highlighting the modifications from the PRF case.

Theorem 2 (Security of StE in PRP). *Let $N = 2^n \geqslant 16$, let $\mathsf{F} : \mathsf{F.Ks} \times \{0,1\}^n \to \{0,1\}^n$ be a keyed permutation family. Let Ext be a 2-universal hash function $\mathsf{h} : \{0,1\}^{kn} \times \{0,1\}^{kn} \to \{0,1\}^{\ell}$. For any S-bounded q-query adversary $\mathcal{A}_{\mathsf{indr}}$, where each query consists of messages of at most B ℓ-bit blocks such that $(S + k(n+1))B \leqslant N/2$, there exists an $(S + B\ell)$-bounded PRP adversary $\mathcal{A}_{\mathsf{prp}}$ (with similar time complexity as $\mathcal{A}_{\mathsf{indr}}$) that issues at most qkB queries to the oracle, such that*

$$\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{StE}[\mathsf{F},k,\mathsf{h}]}(\mathcal{A}_{\mathsf{indr}}) \leqslant \mathsf{Adv}^{\mathsf{prp}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prp}}) + \sqrt{\frac{1}{2} qB\varepsilon} ,$$

where

$$\varepsilon = \frac{\ell}{N^k} + \sum_{t=0}^{k} \binom{k}{t} \left(\frac{(4S + 4kn)B}{N} \right)^t \cdot \min\{\ell, 2^{\ell+1} \cdot (16/N)^{k-t}\} \ .$$

3.2 Proof of Theorem 1

Outline and preliminaries. Most of the proof will consider the StE scheme with direct access to a random function $\mathsf{RF}_{n,n}$. It is immediate to derive a bound when the scheme is instantiated by F at the cost of an additive term $\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prf}})$.

We will be using Lemma 1, applied to a stream consisting of encryptions of the all-zero plaintext (padded to B blocks) or truly random ciphertexts, which we define more formally below. In particular, this will require upper bounding the difference in Shannon entropy (from uniform) of the output of the i-th query, given the adversary's state at that point. As in the proof of the k-XOR construction, we relax our requirements a little, and assume the adversary can generate *arbitrary* S bits of leakage of RF. We will then be using a version of the leftover-hash lemma for bounding Shannon entropy (Proposition 1) to prove the desired bound.

We would naturally need (at the very least) to understand the min-entropy of $V_{i,1} \| \cdots \| V_{i,k}$ conditioned on the state σ_i of stage i. In fact, we will use an even more fine-grained approach, and see $V_{i,1} \| \cdots \| V_{i,k}$ as the convex combination of variables with different levels of entropy. To this end, we will use an approach due to Göös et al. [32] which decomposes a random variable with high min-entropy (in this case, the random function table *conditioned* on σ_i) into a convex combination of (easier to work with) *dense variables*. We use here the definition from [15]:

Definition 1. *A random variable X with range $[M]^N$ is called:*

- *$(1-\delta)$-dense if for every subset $I \subseteq [N]$, the random variable X_I, which is X restricted on coordinates set I, satisfies*

$$\mathsf{H}_\infty(X_I) \geqslant (1-\delta) \cdot |I| \cdot \log M \ .$$

- *$(P, 1-\delta)$-dense if at most P coordinates of X is fixed and X is $(1-\delta)$-dense on the rest coordinates*

Streaming setup. We first define some notations. We use bold-face to denote a vector $\mathbf{R} = (R_1, \ldots, R_k)$. Moreover, we define

$$\mathbf{R}^{\{j\}} = (R_1 + j - 1, R_2 + j - 1, ..., R_k + j - 1) \ ,$$

and $\mathbf{R}^{\{1:j\}} = (\mathbf{R}^{\{1\}}, \mathbf{R}^{\{2\}}, \ldots, \mathbf{R}^{\{j\}})$. For a function F with n-bit inputs, we can further define

$$F[\mathbf{R}^{\{j\}}] := F(0 \parallel R_1 + j - 1) \parallel \cdots \parallel F(k-1 \parallel R_k + j - 1)) \ .$$

Naturally, we extend this to

$$F[\mathbf{R}^{\{1:j\}}] := (F[\mathbf{R}^{\{1\}}], F[\mathbf{R}^{\{2\}}], ..., F[\mathbf{R}^{\{j\}}])$$

Below, we first prove an upper bound for streaming indistinguishability and later upper bound $\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{StE}[\mathsf{RF},k,\mathsf{h}]}$ via the streaming distinguishing advantage. To this end, we define the following two sequences $\mathbf{X} = (X_1, \ldots, X_q)$ and $\mathbf{Y} = (Y_1, \ldots, Y_q)$ of random variables such that:

- $X_i = (W_i, \mathsf{sd}_i, \mathbf{R}_i)$, where $W_i \xleftarrow{\$} \{0,1\}^{B \cdot \ell}$,
- $Y_i = (\mathsf{h}_{\mathsf{sd}_i}(F[\mathbf{R}_i^{\{1\}}]), \ldots, \mathsf{h}_{\mathsf{sd}_i}(F[\mathbf{R}_i^{\{B\}}]), \mathsf{sd}_i, \mathbf{R}_i)$, where F is randomly chosen function from n bits to n bits. (Note that the *same* sampled function is used across all Y_i's.)

In both streams, $\mathsf{sd}_i \xleftarrow{\$} \{0,1\}^s$, and $\mathbf{R}_i = (R_{i,1}, \ldots, R_{i,k})$ is a vector of k random probes. We use L to denote the string length of the stream elements, i.e.,

$$L = |X_i| = |Y_i| = B\ell + s + k(n - \log k) \; .$$

Main lemma. We will use Lemma 1, and rely on the following lemma, which is the core of our analysis.

Lemma 2. *For any S-bounded adversary $\mathcal{A}$ and for all $i \in [q]$,*

$$\mathsf{H}(Y_i \mid \sigma_{i-1}(\mathcal{A}(\mathbf{Y}))) \geqslant L - B\varepsilon$$

where

$$\varepsilon = \frac{\ell}{N^k} + \sum_{t=0}^{k} \binom{k}{t} \left(\frac{(2S + 2kn)B}{N}\right)^t \cdot \min\left\{\ell, 2^{\ell+1}\left(\frac{2}{N}\right)^{k-t}\right\} \; .$$

Proof (of Lemma 2). First, we point out that we can easily find a deterministic function $\mathcal{L}$ such that

$$\mathsf{H}(Y_i \mid \sigma_{i-1}(\mathcal{A}(\mathbf{Y}))) \geqslant \mathsf{H}(Y \mid \mathcal{L}(F)) \; .$$

The function $\mathcal{L}$ is first easily described in randomized form: given F, first simulates the first $i-1$ steps of the interaction of $\mathcal{A}$ with the stream $(Y_1, \ldots, Y_{i-1})$ (by sampling $\mathsf{sd}_1, \ldots, \mathsf{sd}_{i-1}$, as well as $\mathbf{R}_1, \ldots, \mathbf{R}_{i-1}$ itself), and then outputs $\sigma_{i-1}(\mathcal{A}(\mathbf{Y}))$. Then, $\mathcal{L}$ can be made deterministic by fixing the randomness. Therefore, we will now lower bound $\mathsf{H}(Y \mid \mathcal{L}(F))$ for an arbitrary function $\mathcal{L}$.

We now want to better characterize the distribution of F conditioned on $\mathcal{L}(F)$. To this end, we use the following lemma, originally due to Göös *et al* [32], here in a format stated in [14,15].

Lemma 3. *If Γ is a random variable with range $[N]^N$ with min-entropy deficiency $S_\Gamma = n \cdot N - \mathsf{H}_\infty(\Gamma)$, then for every $\delta > 0, \gamma > 0$, Γ can be represented as a convex combination of finitely many $(P, 1-\delta)$-dense variables $\{\Lambda_1, \Lambda_2, ...\}$ for*

$$P = \frac{S_\Gamma + \log 1/\gamma}{\delta \cdot n}$$

and an additional random variable Λ_{end} whose weight is less than γ.

For every $z \in \{0,1\}^S$, we define F_z to be the random function F conditioned on $\mathcal{L}(F) = z$. We define accordingly its min-entropy deficiency $S_z = n \cdot N - \mathsf{H}_\infty(F_z)$. Also, we set $\delta_z = \frac{S_z + \log 1/\gamma}{P \cdot n}$, for some P to be chosen below. By applying Lemma 3, F_z is decomposed into finite number of $(P, 1-\delta_z)$-dense variables $\{\Lambda_{z,1}, \Lambda_{z,2}, \dots\}$, and an additional variable $\Lambda_{z,\mathsf{end}}$ with weight less than γ. We use α_i to denote the weight of each decomposed dense variable in the convex combination. It holds that $\sum_t \alpha_t \geqslant 1 - \gamma$. Also, by the concavity of conditional entropy over probability mass functions,

$$\mathsf{H}(\mathsf{h}_{\mathsf{sd}}(F_z[\mathbf{R}^{\{j\}}]) \mid \mathsf{sd}, \mathbf{R}, F_z[\mathbf{R}^{\{1:j-1\}}]) \\ \geqslant \sum_t \alpha_t \cdot \mathsf{H}(\mathsf{h}_{\mathsf{sd}}(\Lambda_{z,t}[\mathbf{R}^{\{j\}}]) \mid \mathsf{sd}, \mathbf{R}, \Lambda_{z,t}[\mathbf{R}^{\{1:j-1\}}]) \ . \quad (4)$$

It will be sufficient now to give a single entropy lower bound for any variable Λ which is $(P, 1-\delta_z)$-dense, and apply the bound to all $\{\Lambda_{z,1}, \Lambda_{z,2}, \dots\}$. In particular, now note that

$$\mathsf{H}(\mathsf{h}_{\mathsf{sd}}(\Lambda[\mathbf{R}^{\{j\}}]) \mid \mathsf{sd}, \mathbf{R}, \Lambda[\mathbf{R}^{\{1:j-1\}}]) = \mathop{\mathbf{E}}_{\boldsymbol{r}} \left[\mathsf{H}(\mathsf{h}_{\mathsf{sd}}(\Lambda[\boldsymbol{r}^{\{j\}}]) \mid \mathsf{sd}, \Lambda[\boldsymbol{r}^{\{1:j-1\}}]) \right] \\ \geqslant \ell - \mathop{\mathbf{E}}_{\boldsymbol{r}} \left[\min \left\{ \ell, 2^{\ell+1} \cdot 2^{-\mathsf{H}_\infty(\Lambda[\boldsymbol{r}^{\{j\}}] \mid \Lambda[\boldsymbol{r}^{\{1:j-1\}}])} \right\} \right] \ . \quad (5)$$

The last inequality follows from the following version of the Leftover Hash Lemma for Shannon entropy. (We give a proof in the full version for completeness, but note that the proof is similar to that of [10].)

Proposition 1. *If* $\mathsf{h} : \{0,1\}^w \times \{0,1\}^s \to \{0,1\}^\ell$ *is a 2-universal hash function, then for any random variables* $W \in \{0,1\}^w$ *and* Z*, if seed* $\mathsf{sd} \leftarrow \{0,1\}^s$

$$\mathsf{H}(\mathsf{h}_{\mathsf{sd}}(W) \mid \mathsf{sd}, Z) \geqslant \ell - \min\{\ell, 2^{\ell+1} \cdot 2^{-\mathsf{H}_\infty(W|Z)}\} \ .$$

First off, note that

$$\mathsf{H}_\infty(\Lambda[\boldsymbol{r}^{\{j\}}] \mid \Lambda[\boldsymbol{r}^{\{1:j-1\}}]) = -\log \left(\sum_{V \in ([N]^k)^{j-1}} \max_{v \in [N]^k} \mathsf{Pr}\left[\Lambda[\boldsymbol{r}^{\{1:j\}}] = V \parallel v \right] \right)$$

where V enumerates all possible outcome of $\Lambda[\boldsymbol{r}^{\{1:j-1\}}] = (\Lambda[\boldsymbol{r}^{\{1\}}], ..., \Lambda[\boldsymbol{r}^{\{j-1\}}])$, and v iterates over all possible outcome of $\Lambda[\boldsymbol{r}^{\{j\}}]$.

Now, suppose that exactly t probes of $\boldsymbol{r}^{\{j\}}$ hit the P fixed coordinates of Λ and assume that t_0 coordinates of $\boldsymbol{r}^{\{1:j-1\}}$ are fixed. Then, using the fact that Λ is $(1-\delta)$-dense on the remaining $jk - t - t_0$ coordinates, by the union bound, the following inequality holds (the details of calculation can be found in the full version).

$$\log \left(\sum_{V \in ([N]^k)^{j-1}} \max_{v \in [N]^k} \mathsf{Pr}\left[\Lambda[\boldsymbol{r}^{\{1:j\}}] = V \parallel v \right] \right) \leqslant n \left[\delta k(j-1) - (1-\delta)(k-t) \right] \ .$$

Therefore, if t probes of $\boldsymbol{r}^{\{j\}}$ hit the P fixed coordinates of Λ, we have

$$\mathsf{H}_\infty(\Lambda[\boldsymbol{r}^{\{j\}}] \mid \Lambda[\boldsymbol{r}^{\{1:j-1\}}]) \geqslant n\left[(1-\delta)(k-t) - \delta k(j-1)\right] . \tag{6}$$

Now, for $1 \leqslant t \leqslant k$, we let P_t to be the number of fixed coordinates in the domain of t-th probe – in particular, $0 \leqslant P_t \leqslant N/k$ and $\sum_t P_t = P$. Then, let

$$\mu := \mathop{\mathbf{E}}_{\boldsymbol{r}}\left[\min\{\ell, 2^{\ell+1} \cdot 2^{-\mathsf{H}_\infty(\Lambda[\boldsymbol{r}^{\{j\}}] | \Lambda[\boldsymbol{r}^{\{1:j-1\}}])}\}\right]$$

as in (5). Then,

$$\begin{aligned}
\mu &\leqslant \sum_{t=0}^{k} \sum_{U \in \binom{[k]}{t}} \left(\prod_{u \in U} \left(\frac{P_u}{N/k}\right) \prod_{v \notin U} \left(1 - \frac{P_v}{N/k}\right) \min\{\ell, 2^{\ell+1} N^{\delta(j-1)k + (\delta-1)(k-t)}\} \right) \\
&\leqslant \sum_{t=0}^{k} \sum_{U \in \binom{[k]}{t}} \left(\prod_{u \in U} \left(\frac{P_u}{N/k}\right) \cdot \min\{\ell, 2^{\ell+1} \cdot N^{\delta(j-1)k + (\delta-1)(k-t)}\} \right) .
\end{aligned}$$

The above expression is maximized when $P_u = P/k$ for all u. The proof can be found in the full version. Thus we have

$$\begin{aligned}
\mu &\leqslant \sum_{t=0}^{k} \binom{k}{t} \left(\frac{P}{N}\right)^t \cdot \min\{\ell, 2^{\ell+1} \cdot N^{\delta(j-1)k + (\delta-1)(k-t)}\} \\
&= \sum_{t=0}^{k} \binom{k}{t} \left(\frac{P}{N}\right)^t \cdot \min\{\ell, 2^{\ell+1} \cdot 2^{\frac{(S_z + \log(1/\gamma))}{P}(jk-t)} \frac{1}{N^{k-t}}\} =: \nu .
\end{aligned}$$

Plugging this into (4) yields

$$\mathsf{H}(\mathsf{h}_{\mathsf{sd}}(F_z[\mathbf{R}^{\{j\}}]) \mid \mathsf{sd}, \mathbf{R}, F_z[\mathbf{R}^{\{1:j-1\}}]) \geqslant (1-\gamma) \cdot (\ell - \nu) . \tag{7}$$

Next, we will need to take everything in expectation over the sampling of F (and hence of $z = \mathcal{L}(F)$). To this end, we use the following claim to compute $\mathbf{E}_z[\nu]$.

Claim. For any $0 \leqslant t \leqslant k$, $1 \leqslant j \leqslant B$, if $P \geqslant Bk - t$, then it holds that:

$$\mathbf{E}_z[2^{\frac{S_z(jk-t)}{P}}] \leqslant 2^{\frac{S(Bk-t)}{P}} .$$

We left the proof of claim to the full version, but note that the proof is similar to the one from [15]. Now, note that for any function f,

$$\mathbf{E}_z[\min\{\ell, f(z)\}] = \sum_z \mathsf{Pr}[z] \cdot \min\{\ell, f(z)\} \leqslant \min\{\ell, \mathbf{E}_z[f(z)]\} , \tag{8}$$

because $\min\{a,b\} + \min\{c,d\} \leqslant \min\{a+c, b+d\}$ for any a,b,c,d. Using (8), combined with linearity of expectation and the above claim,

$$\begin{aligned}
\mathbf{E}_z[\mu] &\leqslant \sum_{t=0}^{k} \binom{k}{t} \left(\frac{P}{N}\right)^t \cdot \mathbf{E}_z\left[\min\left\{\ell, \frac{2^{\ell+1}\cdot 2^{\frac{(S_z+\log(1/\gamma))}{P}(jk-t)}}{N^{k-t}}\right\}\right] \\
&\leqslant \sum_{t=0}^{k} \binom{k}{t} \left(\frac{P}{N}\right)^t \cdot \min\left\{\ell, 2^{\ell+1}\cdot \mathbf{E}_z\left[\frac{2^{\frac{(S_z+\log(1/\gamma))}{P}(jk-t)}}{N^{k-t}}\right]\right\} \\
&\leqslant \sum_{t=0}^{k} \binom{k}{t} \left(\frac{P}{N}\right)^t \cdot \min\left\{\ell, \frac{2^{\ell+1}\cdot 2^{\frac{(S+\log(1/\gamma))}{P}(Bk-t)}}{N^{k-t}}\right\} .
\end{aligned}$$

Further, we will now finally set $\gamma = N^{-k}$ and $P = (S+kn)B \geqslant Bk$ and simplify this to

$$\begin{aligned}
\mathbf{E}_z[\mu] &\leqslant \sum_{t=0}^{k} \binom{k}{t} \left(\frac{(S+kn)B}{N}\right)^t \cdot \min\left\{\ell, \frac{2^{\ell+1}\cdot 2^k}{N^{k-t}}\right\} \\
&= \sum_{t=0}^{k} \binom{k}{t} \left(\frac{2(S+kn)B}{N}\right)^t \cdot \min\left\{\ell, 2^{\ell+1}\cdot \left(\frac{2}{N}\right)^{k-t}\right\} ,
\end{aligned} \tag{9}$$

because $\frac{S+\log 1/\gamma}{P} \cdot (Bk-t) \leqslant \frac{1}{B}Bk \leqslant k$. Therefore, taking expectations of (7), and using (9), yields

$$\begin{aligned}
&\mathsf{H}(\mathsf{h}_{\mathsf{sd}}(F[\mathbf{R}^{\{j\}}]) \mid \mathsf{sd}, \mathbf{R}, F[\mathbf{R}^{\{1:j-1\}}], \mathcal{L}(F)) \\
&\geqslant (1-\frac{1}{N^k}) \cdot \left(\ell - \sum_{t=0}^{k} \binom{k}{t} \left(\frac{2(S+kn)B}{N}\right)^t \cdot \min\left\{\ell, 2^{\ell+1}\cdot\left(\frac{2}{N}\right)^{k-t}\right\}\right) \\
&\geqslant \ell - \sum_{t=0}^{k} \binom{k}{t} \left(\frac{2(S+kn)B}{N}\right)^t \cdot \min\left\{\ell, 2^{\ell+1}\cdot\left(\frac{2}{N}\right)^{k-t}\right\} - \frac{\ell}{N^k} .
\end{aligned}$$

The proof is concluded by applying chain rule of conditional entropy and obtain

$$\begin{aligned}
&\mathsf{H}(\mathsf{h}_{\mathsf{sd}}(F[\mathbf{R}^{\{1\}}]), ..., \mathsf{h}_{\mathsf{sd}}(F[\mathbf{R}^{\{B\}}]), \mathsf{sd}, \mathbf{R} \mid \mathcal{L}(F)) \\
&= \mathsf{H}(\mathsf{sd}, \mathbf{R} \mid \mathcal{L}(F)) + \mathsf{H}(\mathsf{h}_{\mathsf{sd}}(F[\mathbf{R}^{\{1\}}]), ..., \mathsf{h}_{\mathsf{sd}}(F[\mathbf{R}^{\{B\}}]) \mid \mathsf{sd}, \mathbf{R}, \mathcal{L}(F)) \\
&= L - B\ell + \sum_{j=1}^{B} \mathsf{H}(\mathsf{h}_{\mathsf{sd}}(F[\mathbf{R}^{\{j\}}]) \mid \mathsf{sd}, \mathbf{R}, \mathsf{h}_{\mathsf{sd}}(F[\mathbf{R}^{\{1\}}]), ..., \mathsf{h}_{\mathsf{sd}}(F[\mathbf{R}^{\{j-1\}}]), \mathcal{L}(F)) \\
&\geqslant L - B\left(\sum_{t=0}^{k}\left(\binom{k}{t}\left(\frac{(2S+2kn)B}{N}\right)^t \cdot \min\{\ell, 2^{\ell+1}\cdot(2/N)^{k-t}\}\right) + \frac{\ell}{N^k}\right) .
\end{aligned}$$

□

4 Time-Memory Trade-Off for the k-XOR Construction

In this section, we show that the k-XOR construction (given in Fig. 3), first analyzed by Bellare, Goldreich, and Krawczyk [7] in the memory-independent

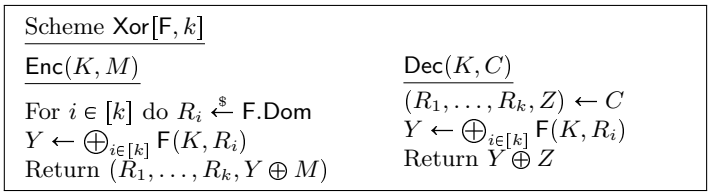

Fig. 3. The k-XOR encryption scheme, $\mathsf{SE} = \mathsf{Xor}[\mathsf{F}, k]$. The key space and message space of SE are $\mathsf{SE.Ks} = \mathsf{F.Ks}$ and $\mathsf{SE.M} = \mathsf{F.Rng}$.

setting, is secure upto $q = (N/S)^{k/2}$ queries for S-bounded adversaries. For the rest of the section, we fix positive integers n and k (required to be even) and let $N = 2^n$.

Theorem 3. *Let* $\mathsf{F} : \mathsf{F.Ks} \times \{0,1\}^n \to \{0,1\}^m$ *be a function family. Let* $\mathsf{SE} = \mathsf{Xor}[\mathsf{F}, k]$ *be the* k*-XOR encryption scheme for some positive integer* k*. Let* $\mathcal{A}_{\mathsf{indr}}$ *be an* S*-bounded INDR-adversary against* SE *that makes at most* q *queries to* Enc*. Then, an* S*-bounded PRF-adversary* $\mathcal{A}_{\mathsf{prf}}$ *can be constructed such that*

$$\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{SE}}(\mathcal{A}_{\mathsf{indr}}) \leqslant \mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prf}}) + 2mq \cdot \sqrt{\left(\frac{4(S+nk)}{N}\right)^k}\,. \tag{10}$$

Moreover, $\mathcal{A}_{\mathsf{prf}}$ *makes at most* $q \cdot k$ *queries to its* Fn *oracle and has running time about that of* $\mathcal{A}_{\mathsf{indr}}$*.*

Discussion of bounds. Our bound supports $q > N$ even with relative small k. Concretely, suppose $S = 2^{80}$ and $N = 2^{128}$. Then for $k = 6$, we can already support upto roughly $q = 2^{(128-80)\cdot(6/2)-8} = 2^{136}$ queries. Note that it does not makes sense to set $q < S$ in our bound. This is because q queries can be stored with $O(q)$ memory. Furthermore, if $q < N/k$, then one can apply the memory independent bound of Bellare, Goldreich, and Krawczyk [7] which is of the form $O(q^2/N^k)$. Hence, our bound really shines when $q \geqslant N$. Lastly, we suspect that our bound is likely not tight in general (it is when $S = O(k \log N)$). In the full version, we show attacks for a broader range of values of S that achieve constant success advantage with $q = O(\left(\frac{N}{S}\right)^k)$.

The above theorem also requires F to be a good PRF – in the full version we discuss how to instantiate it from a block cipher.

Theorem 3 follows from standard hybrid arguments and the single-bit case under random functions, i.e. INDR security of $\mathsf{Xor}[\mathsf{RF}_{n,1}, k]$, which is captured by the following lemma.

Lemma 4. *Let* $\mathsf{SE} = \mathsf{Xor}[\mathsf{RF}_{n,1}, k]$ *be the* k*-XOR encryption scheme for some positive integer* k*. For any* S*-bounded adversary* $\mathcal{A}_{\mathsf{indr}}$ *that makes* q *queries to* Enc*,*

$$\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{SE}}(\mathcal{A}_{\mathsf{indr}}) \leqslant 2q \cdot \sqrt{\left(\frac{4(S+nk)}{N}\right)^k}\,. \tag{11}$$

The proof of Theorem 3 from Lemma 4 consists of standard hybrid arguments (over switching PRF output to random, then over m-output bits to independently random). We shall first prove Lemma 4 and defer the hybrid arguments for later in this section.

BIT-DISTINGUISHING TO BIT-GUESSING. It shall be convenient to consider the following information theoretic quantity $\mathrm{Guess}(\cdot)$, defined for any bit-value random variable B as $\mathrm{Guess}(B) = |2 \cdot \mathsf{Pr}[B = 1] - 1|$. As usual, we extend this to conditioning via $\mathrm{Guess}(B \mid Z) = \mathbf{E}_z\left[\mathrm{Guess}(B \mid Z = z)\right]$. Intuitively, $\mathrm{Guess}(B \mid Z)$ denotes the best possible guessing advantage for bit B, which is also the best bit-distinguishing advantage. Note that if U is a uniform random bit that is independent of Z (B and Z could be correlated), then for any adversary $\mathcal{A}$,

$$\mathsf{Pr}\left[\mathcal{A}(B, Z) \Rightarrow 1\right] - \mathsf{Pr}\left[\mathcal{A}(U, Z) \Rightarrow 1\right] \leqslant \mathrm{Guess}(B \mid Z)\ . \tag{12}$$

Proof of Lemma 4. Consider the INDR games $\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},0}$ and $\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},1}$. We would like to bound

$$\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{SE}}(\mathcal{A}_{\mathsf{indr}}) = \mathsf{Pr}[\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},1}(\mathcal{A}_{\mathsf{indr}})] - \mathsf{Pr}[\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},0}(\mathcal{A}_{\mathsf{indr}})]$$

Towards this end, let us consider hybrid games $\mathsf{H}_0, \ldots, \mathsf{H}_q$ as follows.

Game H_i	$\textsc{Enc}_i(M)$
$F \xleftarrow{\$} \mathsf{Fcs}(\{0,1\}^n, \{0,1\})$	$(R_1, \ldots, R_k) \xleftarrow{\$} (\{0,1\}^n)^k$
$j \leftarrow 0$; $b \xleftarrow{\$} \mathcal{A}^{\textsc{Enc}_i}_{\mathsf{indr}}$	If $j \geqslant i$ then $Z \xleftarrow{\$} \{0,1\}$
Return $b = 1$	Else $Z \leftarrow F(R_1) \oplus \cdots \oplus F(R_k) \oplus M$
	$j \leftarrow j + 1$; Return $(R_1, \ldots, R_k, Z)$

Note that $\mathsf{H}_0 = \mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},0}(\mathcal{A}_{\mathsf{indr}})$ (ideal) and $\mathsf{H}_q = \mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},1}(\mathcal{A}_{\mathsf{indr}})$ (real). Fix some $i \in \{1, \ldots, q\}$. Let $B_i = F(R_{i,1}) \oplus \cdots \oplus F(R_{i,k})$. It holds (by (12)) that

$$\mathsf{Pr}\left[\mathsf{H}_i\right] - \mathsf{Pr}\left[\mathsf{H}_{i-1}\right] \leqslant \mathrm{Guess}(B_i \mid \sigma_{i-1}(\mathcal{A}_{\mathsf{indr}}), (R_{i,1}, \ldots, R_{i,k}))\ , \tag{13}$$

where $\sigma_{i-1}(\mathcal{A}_{\mathsf{indr}})$ is the state of $\mathcal{A}_{\mathsf{indr}}$ right the point where it makes its i-th query to $\textsc{Enc}_i$ (and we assume this query to contain M), and $R_{i,1}, \ldots, R_{i,k}$ are the random inputs generated in that query. Note that $|\sigma_{i-1}(\mathcal{A}_{\mathsf{indr}})| \leqslant S$ and σ_{i-1} is a (randomized-)function of the function table F. However, there must exist a deterministic function $\mathcal{L}_i : \{0,1\}^N \to \{0,1\}^S$, so that

$$\mathrm{Guess}(B_i \mid \sigma_{i-1}(\mathcal{A}_{\mathsf{indr}}), R_{i,1}, \ldots, R_{i,k}) \leqslant \mathrm{Guess}(B_i \mid \mathcal{L}_i(F), R_{i,1}, \ldots, R_{i,k})\ .$$

Hence, to prove Lemma 4, it suffices to show the following lemma.

Lemma 5. *Let $\mathcal{L} : \{0,1\}^N \to \{0,1\}^S$ be any function. Then, for $F \xleftarrow{\$} \{0,1\}^N$, and $R_1, \ldots, R_k \xleftarrow{\$} [N]$,*

$$\mathrm{Guess}(F[R_1] \oplus \cdots \oplus F[R_k] \mid \mathcal{L}(F), R_1, \ldots, R_k) \leqslant 2 \cdot \left(\frac{4(S + nk)}{N}\right)^{k/2}\ . \tag{14}$$

Assuming Lemma 5, we can derive that

$$\begin{aligned}\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{SE}}(\mathcal{A}_{\mathsf{indr}}) &= \sum_{i=0}^{q} \mathsf{Pr}[\mathsf{H}_i] - \mathsf{Pr}[\mathsf{H}_{i-1}] \\ &\leqslant \sum_{i=1}^{q} \mathrm{Guess}(B_i \mid \sigma_{i-1}(\mathcal{A}_{\mathsf{indr}}), R_{i,1}, \ldots, R_{i,k})) \\ &\leqslant \sum_{i=1}^{q} \mathrm{Guess}(B_i \mid \mathcal{L}_i(F), R_{i,1}, \ldots, R_{i,k}) \leqslant 2q \cdot \left(\frac{4(S+nk)}{N}\right)^{k/2} ,\end{aligned}$$

which concludes the proof of Lemma 4. □

Connection to list-decodability of k-XOR code. Lemma 5 is the technical core of our result. Before we go into the details of the proof, we need to recall the definition of list-decoding. Consider the code $\mathsf{k\text{-}XOR} : \{0,1\}^N \to \{0,1\}^{N^k}$, which is defined by

$$\mathsf{k\text{-}XOR}(x)[I] = x[I_1] \oplus \cdots \oplus x[I_k] ,$$

for any $I = (I_1, \ldots, I_k) \in [N]^k$. We say that $\mathsf{k\text{-}XOR} : \{0,1\}^N \to \{0,1\}^{N^k}$ is (ε, L)-list-decodable if for any $z \in \{0,1\}^{N^k}$, there exists at most L codewords within a Hamming ball of radius εN^k around z. The proof of Lemma 5 consists of two steps. First, we translate the left-hand side of (14) in terms of list-decoding properties of k-XOR code. Second, we apply a new list-decoding bound for k-XOR code to obtain (14). We now give some intuition on how Guess relates to list-decoding. First, we fix some deterministic guessing strategy g for $F[R_1] \oplus \cdots \oplus F[R_k]$ given leakage $\mathcal{L}(F)$ and indices $R_1, \ldots, R_k$, which is a function of the form $g : \{0,1\}^S \times [N]^k \to \{0,1\}$ (looking ahead, g shall be fixed to be the "best" one). Note that g can be interpreted as 2^S elements of $\{0,1\}^{N^k}$. In particular, let $g' : \{0,1\}^S \to \{0,1\}^{N^k}$ be the function defined to be

$$g'(x) = g(x, (0, \ldots, 0)) \,\|\, \cdots \,\|\, g(x, (1, \ldots, 1)) .$$

We let G be the set $\{g'(0^S), g'(0^{S-1}1), \ldots, g'(1^S)\}$. Our set G of 2^S guesses lie in the co-domain of the k-XOR code. We now consider a partition of the $\{0,1\}^N$ into sets **Good** and **Bad**, where

$$\begin{aligned}\mathbf{Good} &= \left\{ F \in \{0,1\}^N \mid \nexists z \in G : \mathsf{hw}(\mathsf{k\text{-}XOR}(F), z) \leqslant \left(\frac{1}{2} - \varepsilon/2\right) N^k \right\} , \\ \mathbf{Bad} &= \left\{ F \in \{0,1\}^N \mid \exists z \in G : \mathsf{hw}(\mathsf{k\text{-}XOR}(F), z) \leqslant \left(\frac{1}{2} - \varepsilon/2\right) N^k \right\} .\end{aligned}$$

Note that conditioned on $F \in \mathbf{Good}$, then the guessing strategy g should not achieve advantage better than ε. Using Lemma 6 given below, whose proof shall be given in Section 4.1, we can upper-bound the total number of codewords in **Bad**, as a function of ε.

Lemma 6. *The k-XOR code is* $(\frac{1}{2}-\varepsilon/2, 2^{N-\varepsilon^{2/k}N/4})$*-list decodable, i.e. for any* $z \in \{0,1\}^{N^k}$*, there are at most* $2^{N-\varepsilon^{2/k}N/4}$ *codewords that are within hamming distance* $(\frac{1}{2}-\varepsilon/2)N^k$ *of* z*.*

Finally, obtaining the right-hand size of (14) amounts to picking an ε to minimize $\mathsf{Pr}[F \in \mathbf{Bad}] + \varepsilon$. We proceed to the proof, which formalizes the above intuition.

Proof (of Lemma 5). Consider the code $\mathsf{k\text{-}XOR} : \{0,1\}^N \to \{0,1\}^{N^k}$ defined by

$$\mathsf{k\text{-}XOR}(x)[I] = x[I_1] \oplus \cdots \oplus x[I_k] \ ,$$

for any $I \in [N]^k$. For notational convenience, let $B = F[R_1] \oplus \cdots \oplus F[R_k]$ and $Z = \mathcal{L}(F)$. Consider the following function $Q : \{0,1\}^S \times [N]^k \to [-1,1]$,

$$Q(z, I) = 2 \cdot \mathsf{Pr}\left[B = 1 \mid \mathcal{L}(F) = z, (R_1, \ldots, R_k) = I\right] - 1 \ , \tag{15}$$

where the probability is taken over F. By definition of Guess,

$$\mathrm{Guess}(B \mid \mathcal{L}(F), R_1, \ldots, R_k) = \mathbf{E}\left[|Q(Z, I)|\right] \ , \tag{16}$$

where $Z = \mathcal{L}(F)$ and $I \xleftarrow{\$} [N]^k$. Now, we would like to describe the best guessing strategy $g_z[I]$ for bit B given $\mathcal{L}(F) = z$ and indices I. For each $z \in \{0,1\}^S$, we define $g_z \in \{0,1\}^{N^k}$ as follows. For each $I \in [N]^k$ we let $g_z[I] = 1$ if $Q(z, I) \geqslant 0$ and set $g_z[I] = 0$ otherwise. Intuitively, $g_z[I]$ encodes the best guess for $B = F[I_1] \oplus \cdots F[I_k]$ given that $\mathcal{L}(F) = z$. Hence, for any z and I

$$\frac{1 - |Q(z, I)|}{2} = \mathsf{Pr}\left[B \neq g_{z,I} \mid \mathcal{L}(F) = z, (R_1, \ldots, R_k) = I\right] \ . \tag{17}$$

Taking expectation of both sides over $I \xleftarrow{\$} [N]^k$,

$$\frac{1 - \mathbf{E}\left[|Q(z, I)|\right]}{2} = \mathsf{Pr}\left[B \neq g_{z,I} \mid \mathcal{L}(F) = z\right] = \frac{\mathsf{hw}(\mathsf{k\text{-}XOR}(F) \oplus g_z)}{N^k} \ , \tag{18}$$

where, recall, $\mathsf{hw}(\cdot)$ denotes the hamming weight (number of 1's) of a given string. With slight abuse of notation, we define $Q(z)$ to be

$$Q(z) = \mathbf{E}_{I \xleftarrow{\$} [N]^k}[|Q(z, I)|] = 1 - 2 \cdot \frac{\mathsf{hw}(\mathsf{k\text{-}XOR}(F) \oplus g_z)}{N^k} \ . \tag{19}$$

$Q(z)$ encodes the best possible guessing advantage when $\mathcal{L}(F) = z$, i.e.

$$\mathrm{Guess}(B \mid \mathcal{L}(F), R_1, \ldots, R_k) = \mathbf{E}\left[Q(Z)\right] \ .$$

Define E to be the event that $\mathsf{k\text{-}XOR}(F)$ is of distance more than $(\frac{1}{2}-\varepsilon/2)N^k$ from $g_{\mathcal{L}(F)}$ for some ε to be determined later. Note that given E, then

$$\mathsf{hw}(\mathsf{k\text{-}XOR}(F) \oplus g_{\mathcal{L}(F)}) \geqslant \left(\frac{1}{2} - \varepsilon/2\right) N^k$$

which means that and $Q(\mathcal{L}(F)) \leqslant \varepsilon$. Hence,

$$\begin{aligned}
\mathbf{E}\left[Q(Z)\right] &= \mathsf{Pr}\left[E\right] \cdot \mathbf{E}\left[Q(Z) \mid E\right] + \mathsf{Pr}\left[\neg E\right] \cdot \mathbf{E}\left[Q(Z) \mid \neg E\right] && (20)\\
&\leqslant \varepsilon + \mathsf{Pr}\left[\mathsf{hw}(\mathsf{k\text{-}XOR}(F) \oplus g_{\mathcal{L}(F)}) \leqslant \left(\frac{1}{2} - \varepsilon/2\right) N^k\right] && (21)\\
&\leqslant \varepsilon + \mathsf{Pr}\left[\exists s \in \{0,1\}^S : \mathsf{hw}(\mathsf{k\text{-}XOR}(F) \oplus g_s) \leqslant \left(\frac{1}{2} - \varepsilon/2\right) N^k\right] && (22)\\
&\leqslant \varepsilon + \sum_{s \in \{0,1\}^S} \mathsf{Pr}\left[\mathsf{hw}(\mathsf{k\text{-}XOR}(F) \oplus g_s) \leqslant \left(\frac{1}{2} - \varepsilon/2\right) N^k\right] && (23)\\
&\leqslant \varepsilon + 2^S \cdot 2^{-\varepsilon^{2/k} N/4}\,, && (24)
\end{aligned}$$

where the last equation is by the $((\frac{1}{2} - \varepsilon), 2^{-\varepsilon^{2/k} N/4})$-list decodability of k-XOR-code (Lemma 6). We now set

$$\varepsilon = \sqrt{\left(\frac{4(S+nk)}{N}\right)^k}\,,$$

which makes it so that $\mathbf{E}\left[Q(f(X))\right] \leqslant \varepsilon + 2^{-nk} \leqslant 2 \cdot \varepsilon$. Hence,

$$\mathrm{Guess}(Y \mid f(X), R_1, \ldots, R_k) \leqslant 2 \cdot \left(\frac{4(S+nk)}{N}\right)^{k/2}\,. \qquad (25)$$

This justifies Lemma 5. □

4.1 List Decodability of k-XOR Codes

We relied on the list-decodability of k-XOR code in the proof of Lemma 5. Recall that $\mathsf{k\text{-}XOR} : \{0,1\}^N \to \{0,1\}^{N^k}$ is (ε, L)-list-decodable if for any $z \in \{0,1\}^{N^k}$, there exists at most L codewords within a Hamming ball of radius εN^k around z. The list-decoding property of XOR-code has been studied extensively in complexity theory in the context of hardness amplification. The connection between Yao's XOR Lemma (for a good survey, see [31]) and the list-decodability of XOR-code was first observed by Trevisan [46]. So proofs of hardness amplification results (e.g. [41,34]) using XOR in fact yields algorithmic list-decoding bounds for xor-codes. More recently, [36] has also given approximate list-decoding bounds for k-XOR. We discuss in the full version how the approximate list-decoding bound by [36] can be viewed as (non-approximate) list-decoding bound which lead to an inferior result for the k-XOR construction that promise security upto $q = (N/S)^{k/4}$ instead of $q = (N/S)^{k/2}$. Where as previous works on list-decoding of k-XOR-code focus on algorithmic list-decoding, we are interested in the setting of combinatorial list-decoding, and the best trade-off possible between error ε (especially when it is very close to $1/2$) and the list size L.

Before we begin, we first show the following moment bound on sum of $\{-1,1\}$-valued random variables.

Lemma 7. *Let $F_1, \ldots, F_N$ be i.i.d random variables with $F_i \xleftarrow{\$} \{-1, 1\}$. Then, for any even $m \in \mathbb{N}$*

$$\mathbf{E}\left[\left(\sum_{i\in[N]} F_i\right)^m\right] \leqslant (mN)^{m/2} \ . \tag{26}$$

Proof. Let us first expand the expectation.

$$\mathbf{E}\left[\left(\sum_{i\in[N]} F_i\right)^m\right] = \sum_{I\in[N]^m} \mathbf{E}\left[\prod_{i\in I} F_i\right] \ .$$

We claim that the inside expectation, $\mathbf{E}\left[\prod_{i\in I} F_i\right]$, is either 0 or 1 depending on I. In particular, define I to be even if for every $i \in [N]$, the number of i contained in I is even. First, for any $i \in [N]$, since F_i takes value in $\{-1, 1\}$, it holds that $F_i \cdot F_i = 1$. Hence, observe that $\mathbf{E}\left[\prod_{i\in I} F_i\right]$ is 1 if I is even. Otherwise, if I is not even, we claim that expectation is 0. To see this, suppose i_0 appears an odd number of times in the vector I. We can expand the expectation by conditioning on the value of F_{i_0} being 1 or -1:

$$\mathbf{E}\left[\prod_{i\in I} F_i\right] = \mathbf{E}\left[F_{i_0} \cdot \prod_{i\neq i_0} F_i\right] = \mathbf{E}\left[\prod_{i\neq i_0} F_i\right] - \mathbf{E}\left[\prod_{i\neq i_0} F_i\right] = 0 \ .$$

Therefore,

$$\mathbf{E}\left[\left(\sum_{i\in[N]} F_i\right)^m\right] \leqslant \left|\{I \in [N]^m \mid I \text{ is even }\}\right| \ .$$

For an upper bound of number of even I's, consider the following way of generating even I's. First, we pick a perfect matching (recall that a perfect matching on the complete graph on m vertices is a subset of $m/2$-edges that uses all m vertices) on the complete graph of m-vertices, K_m. Then, for each edge, $e = (v_0, v_1)$, in the matching, we assign a value $i \in [N]$ to nodes v_0 and v_1, i.e. $\ell(v_0) = \ell(v_1) = i$. Now, reading the labels off of each node (wlog we can assume the set of nodes is $[m]$), we obtain an $I = (\ell(0), \ldots, \ell(m-1)) \in [N]^m$ that is even. Note that any even I can be generated in such a way, since given any even I it is easy to find a perfect matching and labeling that results in I.

We move on to compute the number of ways the above can be done. Note that the number of perfect matching is $(m-1) \times (m-3) \times \cdots \times 1$. To see this, let us fix an order of vertices $[m]$, say $1, \ldots, m$. At each step, we shall assign an edge to the smallest vertex that does not yet have an edge. Note that at the i-th step (with i starting at 0), there are exactly $(m - 2i - 1)$ ways to pick the next edge. Hence, the number of perfect matchings on K_m is bounded above by

$$\frac{m!}{2^{m/2}(m/2)!} = \frac{\binom{m}{m/2}}{2^{m/2}} \cdot (m/2)! \leqslant \frac{2^m}{2^{m/2}} \cdot (m/2)^{m/2} \leqslant m^{m/2} \ .$$

Next, for each perfect matching, there are $N^{m/2}$ ways of assigning values to edges, since each one of the $m/2$ edges can be assigned any of the N-values. Hence,

$$\mathbf{E}\left[\left(\sum_{i\in[N]} F_i\right)^m\right] \leqslant (m)^{m/2}\cdot N^{m/2} = (mN)^{m/2}\,.$$

Equipped with Lemma 7, we proceed to prove Lemma 6.

Proof (of Lemma 6). We identify the sets $[N^k]$ with $[N]^k$. Fix some $z \in \{0,1\}^{N^k}$. Let $Z = (Z_1,\ldots,Z_{N^k})$ be the N^k-vector such that $Z_I = (-1)^{z_I}$ for any $I \in [N]^k$. Let $F_1,\ldots,F_n \xleftarrow{\$} \{-1,1\}$. For each $I \in [N]^k$, we define random variable $B_I = \prod_{i\in I} F_i$. Note that if we map B_I to $\{0,1\}$, i.e. define b_I such that $B_I = (-1)^{b_I}$, then $(b_1,\ldots,b_{N^k})$ is just a uniformly random codeword in $\{0,1\}^{N^k}$. We have now that for any $I \in [N^k]$, $(-1)^{b_I\oplus z_I} = Z_I\cdot B_I$. Fix some codeword $(b_1,\ldots,b_{N^k}) \in \{0,1\}^{N^k}$. The hamming distance between it and z is the hamming weight of $s = (b_I \oplus z_I)_{I\in[N]^k}$. Now, note that $\mathsf{hw}(s) \leqslant (1/2-\varepsilon/2)N^k$ if and only if $\sum_I (-1)^{s_I} \geqslant \varepsilon N^k$. Hence, to show that there are at most $2^{N-\varepsilon^{2/k}N/4}$ codewords within radius $(1/2-\varepsilon/2)N^k$ of z, it suffices to show the following bound,

$$\mathsf{Pr}\left[\sum_{I\in[N]^k} Z_I\cdot B_I \geqslant \varepsilon N^k\right] \leqslant 2^{-\varepsilon^{2/k}N/4}\,. \tag{27}$$

Let us compute the p-th moment of $\sum_{I\in[N]^k} Z_I\cdot B_I$ for some even p (we shall fix the particular value of p later).

$$\begin{aligned}
\mathbf{E}\left[\left(\sum_{I\in[N]^k} Z_I\cdot B_I\right)^p\right] &= \mathbf{E}\left[\sum_{I_1,\ldots,I_p} Z_{I_1}\cdots Z_{I_p}B_{I_1}\cdots B_{I_p}\right] &(28)\\
&= \sum_{I_1,\ldots,I_p} (Z_{I_1}\cdots Z_{I_p})\mathbf{E}\left[B_{I_1}\cdots B_{I_p}\right] &(29)\\
&\leqslant \sum_{I_1,\ldots,I_p} \mathbf{E}\left[B_{I_1}\cdots B_{I_p}\right] &(30)\\
&= \mathbf{E}\left[\left(\sum_{I\in[N]^k} B_I\right)^p\right] &(31)\\
&= \mathbf{E}\left[\left(\sum_{i\in[N]} F_i\right)^{k\cdot p}\right] &(32)\\
&\leqslant (kpN)^{kp/2}\,, &(33)
\end{aligned}$$

where (30) is because $\mathbf{E}\left[B_{I_1}\cdots B_{I_p}\right] \in \{0,1\}$ and $Z_{I_1}\cdots Z_{I_p} \in \{-1,1\}$. To see the former claim, compute that

$$\mathbf{E}\left[B_{I_1}\cdots B_{I_p}\right] = \mathbf{E}\left[\prod_{j\in[p]}\prod_{i\in I_j} F_i\right] = \sum_{i\in[N]} \mathbf{E}\left[F_i^{k_i}\right] ,$$

for some $k_1,\ldots,k_N$. Note that $\mathbf{E}\left[F_i^k\right] = 1$ for any even power k, and $\mathbf{E}\left[F_i^k\right] = 0$ for any odd power k. We note that after (30), the expression is *independent* of Z. This is the crucial fact that we rely on when computing the moments of $\sum_{I\in[N]^k} Z_I \cdot B_I$. Applying Markov's inequality to the p-th moment of $\sum_{I\in[N]^k} Z_I \cdot B_I$ and using (33) as well as Lemma 7, we get

$$\mathsf{Pr}\left[\sum_{I\in[N]^k} Z_I\cdot B_I \geqslant \varepsilon N^k\right] \leqslant \frac{(kpN)^{kp/2}}{\varepsilon^p N^{kp}} \leqslant \left(\frac{kp}{\varepsilon^{2/k}N}\right)^{kp/2} . \tag{34}$$

Now, we would be done if we could set p so that $\frac{kp}{\varepsilon^{2/k}N} = \frac{1}{2}$. We cannot do so directly since it only makes sense when p is an even integer. However, we can set $p = p_0$ to be the smallest even integer such that $2kp_0 \geqslant \varepsilon^{2/k}N$. In other words, we set $p = p_0 = 2\cdot\lceil\frac{\varepsilon^{2/k}N}{4k}\rceil$. Note that the right hand side of (34) is minimized when $\frac{kp}{\varepsilon^{2/k}N} = \frac{1}{e}$ and increases as p deviates from this value. Hence, to derive the final bound, as long as $\frac{kp_0}{\varepsilon^{2/k}N} \geqslant \frac{1}{e}$ (which is easily checked), we can plug $p = p_1 = (\varepsilon^{2/k}N)/2k$ into the right-hand side of (34) to derive the final bound of $2^{-\varepsilon^{2/k}N/4}$. □

Acknowledgements Wei Dai was partially supported by grant NSF CNS-1717640. Stefano Tessaro and Xihu Zhang were partially supported by NSF grants CNS-1930117 (CAREER), CNS-1926324, a Sloan Research Fellowship, and a JP Morgan Faculty Award.

References

1. Rohit Agrawal. Samplers and Extractors for Unbounded Functions. In Dimitris Achlioptas and László A. Végh, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019)*, volume 145 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 59:1–59:21, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
2. Benedikt Auerbach, David Cash, Manuel Fersch, and Eike Kiltz. Memory-tight reductions. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part I*, volume 10401 of *LNCS*, pages 101–132. Springer, Heidelberg, August 2017.
3. Yonatan Aumann and Michael O. Rabin. Information theoretically secure communication in the limited storage space model. In Michael J. Wiener, editor, *CRYPTO'99*, volume 1666 of *LNCS*, pages 65–79. Springer, Heidelberg, August 1999.

4. Boaz Barak, Yevgeniy Dodis, Hugo Krawczyk, Olivier Pereira, Krzysztof Pietrzak, François-Xavier Standaert, and Yu Yu. Leftover hash lemma, revisited. In Phillip Rogaway, editor, *CRYPTO 2011*, volume 6841 of *LNCS*, pages 1–20. Springer, Heidelberg, August 2011.
5. Paul Beame, Shayan Oveis Gharan, and Xin Yang. Time-space tradeoffs for learning finite functions from random evaluations, with applications to polynomials. In Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet, editors, *Conference On Learning Theory, COLT 2018, Stockholm, Sweden, 6-9 July 2018*, volume 75 of *Proceedings of Machine Learning Research*, pages 843–856. PMLR, 2018.
6. Mihir Bellare and Wei Dai. Defending against key exfiltration: Efficiency improvements for big-key cryptography via large-alphabet subkey prediction. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, *ACM CCS 2017*, pages 923–940. ACM Press, October / November 2017.
7. Mihir Bellare, Oded Goldreich, and Hugo Krawczyk. Stateless evaluation of pseudorandom functions: Security beyond the birthday barrier. In Michael J. Wiener, editor, *CRYPTO'99*, volume 1666 of *LNCS*, pages 270–287. Springer, Heidelberg, August 1999.
8. Mihir Bellare, Daniel Kane, and Phillip Rogaway. Big-key symmetric encryption: Resisting key exfiltration. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 373–402. Springer, Heidelberg, August 2016.
9. Mihir Bellare and Phillip Rogaway. The security of triple encryption and a framework for code-based game-playing proofs. In Serge Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 409–426. Springer, Heidelberg, May / June 2006.
10. C. H. Bennett, G. Brassard, C. Crepeau, and U. M. Maurer. Generalized privacy amplification. *IEEE Transactions on Information Theory*, 41(6):1915–1923, Nov 1995.
11. Andrej Bogdanov, Manuel Sabin, and Prashant Nalini Vasudevan. XOR codes and sparse learning parity with noise. In Timothy M. Chan, editor, *30th SODA*, pages 986–1004. ACM-SIAM, January 2019.
12. Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger. Biclique cryptanalysis of the full AES. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 344–371. Springer, Heidelberg, December 2011.
13. Larry Carter and Mark N. Wegman. Universal classes of hash functions. *J. Comput. Syst. Sci.*, 18(2):143–154, 1979.
14. Sandro Coretti, Yevgeniy Dodis, and Siyao Guo. Non-uniform bounds in the random-permutation, ideal-cipher, and generic-group models. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part I*, volume 10991 of *LNCS*, pages 693–721. Springer, Heidelberg, August 2018.
15. Sandro Coretti, Yevgeniy Dodis, Siyao Guo, and John P. Steinberger. Random oracles and non-uniformity. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part I*, volume 10820 of *LNCS*, pages 227–258. Springer, Heidelberg, April / May 2018.
16. Wei Dai, Viet Tung Hoang, and Stefano Tessaro. Information-theoretic indistinguishability via the chi-squared method. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part III*, volume 10403 of *LNCS*, pages 497–523. Springer, Heidelberg, August 2017.

17. Anindya De and Luca Trevisan. Extractors using hardness amplification. In Irit Dinur, Klaus Jansen, Joseph Naor, and José D. P. Rolim, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 12th International Workshop, APPROX 2009, and 13th International Workshop, RANDOM 2009, Berkeley, CA, USA, August 21-23, 2009. Proceedings*, volume 5687 of *Lecture Notes in Computer Science*, pages 462–475. Springer, 2009.
18. Giovanni Di Crescenzo, Richard J. Lipton, and Shabsi Walfish. Perfectly secure password protocols in the bounded retrieval model. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 225–244. Springer, Heidelberg, March 2006.
19. Itai Dinur. On the streaming indistinguishability of a random permutation and a random function. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part II*, volume 12106 of *LNCS*, pages 433–460. Springer, Heidelberg, May 2020.
20. Itai Dinur. Tight time-space lower bounds for finding multiple collision pairs and their applications. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part I*, volume 12105 of *LNCS*, pages 405–434. Springer, Heidelberg, May 2020.
21. Stefan Dziembowski. Intrusion-resilience via the bounded-storage model. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 207–224. Springer, Heidelberg, March 2006.
22. Stefan Dziembowski and Ueli M. Maurer. Tight security proofs for the bounded-storage model. In *34th ACM STOC*, pages 341–350. ACM Press, May 2002.
23. Stefan Dziembowski and Ueli M. Maurer. Optimal randomizer efficiency in the bounded-storage model. *Journal of Cryptology*, 17(1):5–26, January 2004.
24. Sumegha Garg, Pravesh K. Kothari, and Ran Raz. Time-space tradeoffs for distinguishing distributions and applications to security of goldreich's PRG. *CoRR*, abs/2002.07235, 2020.
25. Sumegha Garg, Ran Raz, and Avishay Tal. Extractor-based time-space lower bounds for learning. In Ilias Diakonikolas, David Kempe, and Monika Henzinger, editors, *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 990–1002. ACM, 2018.
26. Peter Gaži. Plain versus randomized cascading-based key-length extension for block ciphers. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part I*, volume 8042 of *LNCS*, pages 551–570. Springer, Heidelberg, August 2013.
27. Peter Gaži and Stefano Tessaro. Efficient and optimally secure key-length extension for block ciphers via randomized cascading. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 63–80. Springer, Heidelberg, April 2012.
28. Ashrujit Ghoshal, Joseph Jaeger, and Stefano Tessaro. The memory-tightness of authenticated encryption. In Daniele Micciancio and Thomas Ristenpart, editors, *CRYPTO 2020, Part I*, volume 12170 of *LNCS*, pages 127–156. Springer, Heidelberg, August 2020.
29. Ashrujit Ghoshal and Stefano Tessaro. On the memory-tightness of hashed ElGamal. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part II*, volume 12106 of *LNCS*, pages 33–62. Springer, Heidelberg, May 2020.
30. Oded Goldreich. Candidate one-way functions based on expander graphs. Cryptology ePrint Archive, Report 2000/063, 2000. http://eprint.iacr.org/2000/063.
31. Oded Goldreich, Noam Nisan, and Avi Wigderson. On yao's xor-lemma. In *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, pages 273–301. Springer, 2011.

32. Mika Göös, Shachar Lovett, Raghu Meka, Thomas Watson, and David Zuckerman. Rectangles are nonnegative juntas. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *47th ACM STOC*, pages 257–266. ACM Press, June 2015.
33. Viet Tung Hoang and Stefano Tessaro. Key-alternating ciphers and key-length extension: Exact bounds and multi-user security. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 3–32. Springer, Heidelberg, August 2016.
34. Russell Impagliazzo. Hard-core distributions for somewhat hard problems. In *36th FOCS*, pages 538–545. IEEE Computer Society Press, October 1995.
35. Russell Impagliazzo, Ragesh Jaiswal, and Valentine Kabanets. Approximately list-decoding direct product codes and uniform hardness amplification. In *47th FOCS*, pages 187–196. IEEE Computer Society Press, October 2006.
36. Russell Impagliazzo, Ragesh Jaiswal, and Valentine Kabanets. Approximate list-decoding of direct product codes and uniform hardness amplification. *SIAM Journal on Computing*, 39(2):564–605, 2009.
37. Russell Impagliazzo, Ragesh Jaiswal, Valentine Kabanets, and Avi Wigderson. Uniform direct product theorems: simplified, optimized, and derandomized. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 579–588. ACM Press, May 2008.
38. Russell Impagliazzo, Leonid A. Levin, and Michael Luby. Pseudo-random generation from one-way functions (extended abstracts). In *21st ACM STOC*, pages 12–24. ACM Press, May 1989.
39. Joseph Jaeger and Stefano Tessaro. Tight time-memory trade-offs for symmetric encryption. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part I*, volume 11476 of *LNCS*, pages 467–497. Springer, Heidelberg, May 2019.
40. Gillat Kol, Ran Raz, and Avishay Tal. Time-space hardness of learning sparse parities. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *49th ACM STOC*, pages 1067–1080. ACM Press, June 2017.
41. Leonid A Levin. One way functions and pseudorandom generators. *Combinatorica*, 7(4):357–363, 1987.
42. Ueli M. Maurer. Conditionally-perfect secrecy and a provably-secure randomized cipher. *Journal of Cryptology*, 5(1):53–66, January 1992.
43. Ran Raz. Fast learning requires good memory: A time-space lower bound for parity learning. In Irit Dinur, editor, *57th FOCS*, pages 266–275. IEEE Computer Society Press, October 2016.
44. Ran Raz. A time-space lower bound for a large class of learning problems. In Chris Umans, editor, *58th FOCS*, pages 732–742. IEEE Computer Society Press, October 2017.
45. Stefano Tessaro and Aishwarya Thiruvengadam. Provable time-memory trade-offs: Symmetric cryptography against memory-bounded adversaries. In Amos Beimel and Stefan Dziembowski, editors, *TCC 2018, Part I*, volume 11239 of *LNCS*, pages 3–32. Springer, Heidelberg, November 2018.
46. Luca Trevisan. List-decoding using the XOR lemma. In *44th FOCS*, pages 126–135. IEEE Computer Society Press, October 2003.
47. Salil P. Vadhan. On constructing locally computable extractors and cryptosystems in the bounded storage model. In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 61–77. Springer, Heidelberg, August 2003.
48. Yuyu Wang, Takahiro Matsuda, Goichiro Hanaoka, and Keisuke Tanaka. Memory lower bounds of reductions revisited. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part I*, volume 10820 of *LNCS*, pages 61–90. Springer, Heidelberg, April / May 2018.