Tight Time-Memory Trade-offs for Symmetric Encryption

Joseph Jaeger[1] and Stefano Tessaro[2]

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

Abstract. Concrete security proofs give upper bounds on the attacker's advantage as a function of its time/query complexity. Cryptanalysis suggests however that other resource limitations – most notably, the attacker's memory – could make the achievable advantage smaller, and thus these proven bounds too pessimistic. Yet, handling memory limitations has eluded existing security proofs.

This paper initiates the study of time-memory trade-offs for basic symmetric cryptography. We show that schemes like counter-mode encryption, which are affected by the Birthday Bound, become *more secure* (in terms of time complexity) as the attacker's memory is reduced.

One key step of this work is a generalization of the Switching Lemma: For adversaries with S bits of memory issuing q distinct queries, we prove an n-to-n bit random function indistinguishable from a permutation as long as $S \times q \ll 2^n$. This result assumes a combinatorial conjecture, which we discuss, and implies right away trade-offs for deterministic, stateful versions of CTR and OFB encryption.

We also show an unconditional time-memory trade-off for the security of *randomized* CTR based on a secure PRF. Via the aforementioned conjecture, we extend the result to assuming a PRP instead, assuming only one-block messages are encrypted.

Our results solely rely on standard PRF/PRP security of an underlying block cipher. We frame the core of our proofs within a general framework of indistinguishability for streaming algorithms which may be of independent interest.

Keywords: Provable security, symmetric cryptography, time-memory trade-offs

1 Introduction

Concrete security proofs upper bound the adversarial advantage ε as a function of the adversary's *resources*. A scheme is deemed secure if the advantage is small for all feasible resource amounts. The classical approach captures such resources in terms of *running time* and/or *description size*.

Time is however not the only resource to determine feasibility of an attack. In particular, the *memory* costs also matter – in the context of provable security, these were first studied by Auerbach et al. [4] and Wang et al. [26], who considered the tightness of reductions with respect to memory usage. Memory-tight reductions lift an assumed time-memory trade-off for the assumption to one for the scheme, and this is particularly important when the underlying assumption does not admit low-memory attacks (e.g., this is true for the LPN problem).

Earlier work on time-memory tradeoffs in symmetric cryptography focused on cryptanalytic attacks [15,5] or precomputation attacks against primitives like hash functions [6].

Symmetric cryptography. Memory tightness is less useful for symmetric cryptography: A typical assumption here is that AES is a PRP for attackers with large time complexity, e.g., $T = 2^{100}$, but the best generic attack is memoryless, so there is generally no trade-off to be assumed.

Still, time-memory trade-offs may affect the actual *modes of operation*. For example, it is well known that (randomized) counter mode (CTR\$) allows to encrypt no more than $q = \sqrt{N}$ plaintexts when using an n-bit block cipher (here, $N = 2^n$), yet restricting memory to only store S bits may help. Indeed, let the i-th message m_i be encrypted as $(r_i, c_i = \mathrm{AES}_K(r_i) \oplus m_i)$, where r_i is a random string. The *optimal* distinguishing attack waits for $r_i = r_j$ to occur for $i \neq j$, in which case $c_i \oplus c_j = m_i \oplus m_j$ – which is unlikely to hold if c_i and c_j are random. But this also requires remembering approximately $\sqrt{N}$ r_i's. If we can only store fewer of them, then we need a collision with one of the r_i's we remember, and the attack advantage decrease to $\frac{Sq}{N}$ when q messages are encrypted. However, is this attack the optimal one? – a proof would have to argue *over all possible* adversarial strategies storing S bits of partial information.

Remarkably, despite schemes like CTR\$ being decades old, the question of proving bounds that take memory into account has remained open.

Our results: Overview. This paper takes a ground-up approach to *proving* time-memory trade-offs. To this end, we start with exactly those simple symmetric encryption schemes like CTR\$ and OFB we ought to understand, and develop proofs and proof techniques – mostly relying on information-theoretic and combinatorial tools – aimed at showing that conjectured trade-offs are optimal.

A common trait of basic encryption schemes is that they are only secure up to the Birthday Bound. For stateless, randomized schemes, this is because inputs to the block cipher are otherwise going to repeat. Also, even when inputs *are* distinct, non-repeating block-cipher outputs become easily distinguishable from random. We will show that this fact is no longer valid if the adversary's memory capacity does *not* exceed $\sqrt{N}$, and more generally, we show a trade-off between the number of encryptions and the attacker's memory.

For example, we revisit the well-known Switching Lemma in the memory-bounded setting: under a combinatorial conjecture (see details below), we show that an adversary making T *distinct* queries to a random function or a random permutation cannot tell them apart with advantage larger than $O(\sqrt{ST/N})$. The special case $S = T$ is the original switching lemma. This gives us bounds for stateful CTR and OFB, assuming the underlying block cipher is a sufficiently secure PRP. We consider the question fundamental enough to justify a partial

Scheme	Underlying Primitive	Bound
CTR	PRF	$\varepsilon_{\mathsf{prf}}$
	PRP	$\varepsilon_{\mathsf{prp}} + \mathcal{O}_{\mathrm{sl}}(T, S, N)$
OFB	PRF	Insecure when $T \in \Omega(\sqrt{N})$
	PRP	$\varepsilon_{\mathsf{prp}} + \mathcal{O}_{\mathrm{sl}}(T, S, N) + O(T/N)$
$\mathsf{CTR\$}$	PRF	$\varepsilon_{\mathsf{prf}} + O(\sqrt{ST/N})$
1-block $\mathsf{CTR\$}$	weak-PRP	$\varepsilon_{\mathsf{wprp}} + 3\mathcal{O}_{\mathrm{sl}}(T, S, N)$
Encrypt-then-PRF	INDR and weak-PRF	$\varepsilon_{\mathsf{indr}} + \varepsilon_{\mathsf{wprf}} + O(\sqrt{ST/N})$

Fig. 1. Encryption schemes we analyze. Schemes with a \$ are randomized, otherwise they are deterministic. If Conjecture 1 holds then $\mathcal{O}_{\mathrm{sl}}(T, S, N) \in O(\sqrt{ST/N})$. Bounds are for INDR security. S is the memory bound of the adversary, T is the number of blocks encrypted, and N is the domain size of the family of functions.

answer even under a conjecture – moreover, the reduction to this conjecture is highly non-trivial, and a failure of the conjecture is likely to only minimally impact this bound.

We also show a bound of $O(\sqrt{ST\ell/N})$ for randomized $\mathsf{CTR\$}$ based on a pseudorandom *function* (PRF), where ℓ is a bound on the number of blocks per encrypted message. This result does not need any conjecture, beyond PRF security. For the case $\ell = 1$, we show that under the aforementioned conjecture, the result holds when the scheme is based on a PRP, instead of a PRF.

An overview of our results for encryptions schemes is given in Figure 1. We discuss them in more detail below, but first address an important piece of recent related work.

Related work. It is worth noting that our work complements a recent paper by Tessaro and Thiruvengadam [25]. Their goal are schemes with security *as high as possible*, well beyond 2^n (where n is the block length of the cipher), provided the cipher is secure enough (e.g., it has a long key), and adversarial memory is bounded. In their work, neither tightness nor practical efficiency is a concern. Here, in contrast, we focus on *tightness* for simple, deployed cryptography. As a result of this, we end up facing different, and somewhat more technically challenging problems.

A framework: Streaming indistinguishability. The common denominator of our security proofs is that they reduce to a new, yet natural, setting of memory-bounded streaming algorithms which we refer to as *streaming indistinguishability.* In essence, a memory-bounded algorithm $\mathcal{A}$ is given access, one value at a time, to one of two streams

$$X_1, X_2, \ldots \quad \text{or} \quad Y_1, Y_2, \ldots\ ,$$

with different distributions. The goal is to distinguish them.

To the best of our knowledge, the existing literature on streaming algorithms does not consider this problem explicitly. Rather, the focus is mostly on worstcase complexity (we care about average-case), and search problems. However, one can cast classical problems like building PRGs against space-bounded read-once branching programs (cf. e.g. [21]), as a special case of this setting, where the X_i's are the output bits of the PRG and the Y_i's are random bits.

THE SWITCHING LEMMA. Let us first address our generalized Switching Lemma. It is well known that the advantage of a T-query distinguisher $\mathcal{A}$ trying to tell apart a truly random permutation P from a truly random function F (both from n bits to n bits) is at most T^2/N, which is tight. Also, an optimal distinguisher making $T \approx \sqrt{N}$ can be implemented to only use $S \ll \sqrt{N}$ bits, e.g., with the help of a memory-less collision-finding algorithms (e.g., using Pollard's ρ-method [23,24]). One uses the fact that when accessing P, the algorithm will never succeed in finding a collision.

One observation, however, is that in many useful scenarios, the resulting $\mathcal{A}$ never queries the same input *twice* and it is not hard to see that any memory-less collision-finding strategy *will* query the same input twice.

We show that, assuming the validity of a conjecture we explain next, under non-repeating queries, the Switching Lemma indeed holds with a tradeoff of the form $S \times T = N$. In fact, we prove a more general (and also fundamental) statement about the advantage of distinguishing two streams: The first, $X_1, X_2, \ldots$ samples n-bit values with replacement, the second, $Y_1, Y_2, \ldots$, without.

A CONJECTURE. A proof of a non-trivial bound appears out of reach. Instead, we give a proof that relies on a (plausible) combinatorial conjecture involving *hypergraphs*.

Recall that a k-hypergraph with N vertices is a collection $H = \{e_1, \ldots, e_m\}$, where the e_i's are distinct size-k subsets of $[N] = \{1, 2, \ldots, N\}$. The *degree* $d_H(i)$ of $i \in [N]$ is the number of e_j's such that $i \in e_j$. Then, we look at the maximum $D^2(m)$, over all m-edge hypergraphs H, of the function

$$D^2(H) = \sum_{i=1}^{N} d_H(i)^2 \ .$$

Estimating $D^2(m)$ is challenging: The only known upper bound [9] is loose, and the general question is believed to be out of reach [16]. This is because degree sequences of hypergraphs are poorly understood, even more so when restricted to m edges. Only for the special case of graphs (i.e., $k = 2$) is the question well understood (cf. e.g. [14,10,20,1]), though far from trivial.

Our conjecture will be on the value of $D^2(m)$ when $k > N/2$ for *specific values of* m. We will assume in particular that if $m = \binom{A}{k}$, then the complete hypergraph containing all k-element subsets of $\{1, \ldots, A\}$ achieves $D^2(m)$. We stress that even a slight relaxation of this conjecture would only affect our proof slightly.

RANDOMIZED COUNTER MODE. The above switching lemma for distinct inputs only applies to stateful schemes. Let us look now instead at randomized $\mathsf{CTR\$}$ described above and, for simplicity, let us assume that we encrypt single-block plaintexts. Assuming the underlying block cipher is a PRF, the resulting security game can again be cast as a streaming (in)distinguishability setting with

$$X_i = (R_i, Z_i)\ , \qquad Y_i = (R_i, F(R_i))\ ,$$

where F is a random function from n bits to n bits and the R_i, Z_i's are random, independent n-bit strings. We will show a bound of $O(\sqrt{ST/N})$. Interesting, once cast in the right language, the proof is fairly elementary and uses only simple properties of Shannon entropies – what is novel here is the usage of these tools to prove the security of symmetric cryptography, and the fact that they are robust to dealing with memory restrictions.

In practice, of course, F is more likely to be a permutation, as it is built from a block cipher. However, our proof techniques seems not to extend directly to random permutations. We also cannot apply the Switching Lemma *directly*, because R_i's will not be distinct.

We will however do something different – we will apply the streaming indistinguishability result underlying the Switching Lemma to the R_i's first, telling us they can be replaced by random, distinct ones when encrypting single-block plaintexts. This will allow us to ultimately to replace F with a permutation – again by the Switching Lemma – but for a concrete bound, we will need to resort, again to our conjecture. (This can be thought, more generally, as extending the Switching Lemma to the case of random inputs.)

We could of course build a beyond-birthday secure PRF from a block cipher directly, e.g., using the xor construction [7,22,12], but this would require two block-cipher calls per block, or Iwata's CENC [17,18] for better amortized efficiency. We note that we also apply these techniques to analyze the confidentiality of Encrypt-then-PRF.

Outline of this paper. Section 2 introduces notation and provides necessary information theoretic and cryptographic background. Section 3.1 introduces our general streaming setting. Section 3.2 and Section 4.1 introduce our main streaming theorems which are proven in Section 3.3 and Section 4.2, respectively. In Section 3.4 and Section 4.3 we apply these respective theorems to cryptographic reductions. We emphasize that while the analysis in Section 3 requires a conjecture, the results of Section 4 are unconditional.

2 Definitions

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

If $M \in \{0,1\}^*$ is a string, then $|M|$ denotes its bitlenth. If $m \in \mathbb{N}$ and $M \in (\{0,1\}^m)^*$, then $|M|_m = |M|/m$ denotes the blocklength 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 empty string is ε.

Algorithms are randomized when not specified otherwise. If $\mathcal{A}$ is an algorithm, then $y \leftarrow \mathcal{A}^{\mathrm{O}_1,\cdots}(x_1,\ldots;r)$ denotes running $\mathcal{A}$ on inputs $x_1,\ldots$ and coins r with access to oracles $\mathrm{O}_1,\ldots$ to produce output y. The notation $y \xleftarrow{\$} \mathcal{A}^{\mathrm{O}_1,\cdots}(x_1,\ldots)$ denotes picking r at random then running $y \leftarrow \mathcal{A}^{\mathrm{O}_1,\cdots}(x_1,\ldots;r)$. 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).

Our cryptographic reductions will use pseudocode games (inspired by the code-based framework of [8]). See Fig. 2 for some example games. We let $\mathsf{Pr}[\mathsf{G}]$ denote the probability that game G outputs `true`. The model underlying this pseudocode is the following formalism

2.1 Model of computation

Computational Model. Our model is based on those of [2,3,25]. We consider a space-bounded adversary interacting with an oracle O.

The interaction between an adversary and oracle occurs over q stages. In the i-th stage, the adversary deterministically computes, as a function of the state σ_{i-1} and stage number i, a query x_i to O.[3] Then the adversary is give $y_i = \mathrm{O}(x_i)$ (with the same inputs as before) based on which it computes the next state σ_i. The state σ_0 is fixed and defined by $\mathcal{A}$. The final output of $\mathcal{A}$ is σ_q. In code, stage i behaves as follows, **Stage** i: $x_i \leftarrow \mathcal{A}(i,\sigma_{i-1})$; $y_i \leftarrow \mathrm{O}(x_i)$; $\sigma_i \xleftarrow{\$} \mathcal{A}(i,\sigma_{i-1},y_i)$.

Complexity Measures. An adversary $\mathcal{A}$ is S-*bounded* if $|\sigma_i| \leqslant S$ holds for all i. The running time of $\mathcal{A}$ is T if it queries at most T bits to its oracle. These complexity measures do not count the local state or time used by $\mathcal{A}$ *during* a round. This strengthens our main proofs which are information theoretic in nature and only require that the states σ_i and T are bounded in size.

Our applications of these main proofs will involve cryptographic reductions. These complexity measures are not appropriate for this because they could hide a weakness in a reduction that "cheats" by using much more local state or computation time during a round. None of our reductions have such a weakness so we leave reduction efficiency claims informal. See [4] for discussion of what conventions should be used for measuring the memory complexity of a reduction. Our reductions are given via explicit pseudocode so their complexity with respect to particular conventions can easily be extracted.

2.2 Information-theoretic preliminaries

Entropies and KL-divergence. For probability distributions $P, Q : \mathcal{X} \to [0,1]$ where $Q(x) > 0$ for all $x \in \mathcal{X}$, the Shannon and collision entropies are

$$H(P) = -\sum_{x\in\mathcal{X}} P(x)\log(P(x)) \text{ and } H_2(P) = -\log\left(\sum_{x\in\mathcal{X}} P(x)^2\right) .$$

[3] We insist on this computation being deterministic for convenience and because we can think of x_i having been included as part of σ_{i-1}.

Statistical distance and KL-divergence are defined by

$$\mathsf{SD}(P,Q) = \frac{1}{2}\sum_{x\in\mathcal{X}} |P(x) - Q(x)| \text{ and } \mathsf{KL}(P\|Q) = \sum_{x\in\mathcal{X}} P(x)\log\left(\frac{P(x)}{Q(x)}\right) .$$

Pinsker's inequality says that $\mathsf{SD}(P,Q) \leqslant \sqrt{\mathsf{KL}(P\|Q)/2}$.

As usual, for two random variables X and Y with distributions P_X and P_Y, we write $\mathsf{KL}(X\|Y)$ for $\mathsf{KL}(P_X\|P_Y)$ (and the analogous notation for H and H_2).

Lemma 1. *Let X, Y be random variables with range $\mathcal{X}$ with $\Pr[X = x] > 0$ for all $x \in \mathcal{X}$. Let $F : \mathcal{X} \to \{0,1\}^*$ be a (possibly randomized) function. Then,*

$$\mathsf{KL}(F(X)\|F(Y)) \leqslant \mathsf{KL}(X\|Y) .$$

Proof. For compactness, denote $\mathsf{P}_Z(x) = \Pr[Z = x]$ for any random variable Z. First, we note that we can consider without loss of generality deterministic F's. Indeed, by convexity (cf. e.g. [11]),

$$\mathsf{KL}(F(X)\|F(Y)) \leqslant \sum_f \Pr[F = f] \cdot \mathsf{KL}(f(X)\|f(Y)) .$$

Now fix a function $f : \mathcal{X} \to \{0,1\}^*$. From the log-sum inequality we obtain

$$\begin{aligned}
\mathsf{KL}(F(X)\|F(Y)) &= \sum_z \mathsf{P}_{F(X)}(z)\log\left(\frac{\mathsf{P}_{F(X)}(z)}{\mathsf{P}_{F(Y)}(z)}\right) \\
&= \sum_z \left(\sum_{x\in f^{-1}(z)} \mathsf{P}_X(x)\right)\cdot\log\left(\frac{\sum_{x\in f^{-1}(z)}\mathsf{P}_X(x)}{\sum_{x\in f^{-1}(z)}\mathsf{P}_Y(x)}\right) \\
&\leqslant \sum_z \sum_{x\in f^{-1}(z)} \mathsf{P}_X(x)\log\left(\frac{\mathsf{P}_X(x)}{\mathsf{P}_Y(x)}\right) \\
&= \sum_{x\in\mathcal{X}} \mathsf{P}_X(x)\log\left(\frac{\mathsf{P}_X(x)}{\mathsf{P}_Y(x)}\right) .
\end{aligned}$$

The last equality follows because every x is the pre-image of *exactly* one z. □

2.3 Cryptographic preliminaries

FAMILY OF FUNCTIONS. A family of functions F specifies algorithms $\mathsf{F.K}$ and $\mathsf{F.Ev}$ (where the latter of these is deterministic) and sets $\mathsf{F.Dom}$ and $\mathsf{F.Rng}$. Key generation algorithm $\mathsf{F.K}$ takes no input and outputs a key K. Evaluation algorithm takes as input key K and $X \in \mathsf{F.Dom}$ to return $Y \in \mathsf{F.Rng}$. We write $K \xleftarrow{\$} \mathsf{F.K}$ and $Y \leftarrow \mathsf{F.Ev}(K, X)$.

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

Game $\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},b}(\mathcal{A})$	Game $\mathsf{G}^{\mathsf{prp}}_{\mathsf{F},b}(\mathcal{A})$	Game $\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},b}(\mathcal{A})$
$K \xleftarrow{\$} \mathsf{F.K}$	$K \xleftarrow{\$} \mathsf{F.K}$	$\sigma \xleftarrow{\$} \mathsf{SE.Sg}$
$F \xleftarrow{\$} \mathsf{Fcs}(\mathsf{F.Dom}, \mathsf{F.Rng})$	$P \xleftarrow{\$} \mathsf{Perm}(\mathsf{F.Dom})$	$b' \xleftarrow{\$} \mathcal{A}^{\textsc{Enc}}$
$b' \xleftarrow{\$} \mathcal{A}^{\textsc{Ror}}$	$b' \xleftarrow{\$} \mathcal{A}^{\textsc{Ror}}$	Return $b' = 1$
Return $b' = 1$	Return $b' = 1$	
$\textsc{Ror}(X)$	$\textsc{Ror}(X)$	$\textsc{Enc}(M)$
$Y_1 \leftarrow \mathsf{F.Ev}(K, X)$	$Y_1 \leftarrow \mathsf{F.Ev}(K, X)$	$(\sigma, C_1) \leftarrow \mathsf{SE.E}(\sigma, M)$
$Y_0 \leftarrow F(X)$	$Y_0 \leftarrow P(X)$	$C_0 \xleftarrow{\$} \{0,1\}^{\lvert M\rvert + \mathsf{SE.xl}}$
Return Y_b	Return Y_b	Return C_b

Fig. 2. Security games for PRF/PRP security of a family of functions (Left/Middle) and INDR security of an encryption scheme (Right).

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

Let F be a family of functions. PRF security requires that $\mathsf{F.Ev}(K, \cdot)$ looks like a truly random function to somebody who does not know K. Consider the game $\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},b}(\mathcal{A})$ shown on the left side of Figure 2. It it parameterized by F, a bit $b \in \{0,1\}$, and an adversary. The adverasry is given access to an oracle $\textsc{Ror}$ which on input X either returns F applied to X ($b = 1$) or the output of a random function on X ($b = 0$). The advantage of $\mathcal{A}$ against F is defined by $\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}) = \Pr[\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},1}(\mathcal{A})] - \Pr[\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},0}(\mathcal{A})]$.

PRP security of a blockcipher F is defined analogously by the game $\mathsf{G}^{\mathsf{prp}}_{\mathsf{F},b}(\mathcal{A})$ shown in the middle of Figure 2. This is essentially the same except the random function $F \in \mathsf{Fcs}(\mathsf{F.Dom}, \mathsf{F.Rng})$ has been replaced by a random permutation $P \in \mathsf{Perm}(\mathsf{F.Dom})$. The advantage of $\mathcal{A}$ against F is defined by $\mathsf{Adv}^{\mathsf{prp}}_{\mathsf{F}}(\mathcal{A}) = \Pr[\mathsf{G}^{\mathsf{prp}}_{\mathsf{F},1}(\mathcal{A})] - \Pr[\mathsf{G}^{\mathsf{prp}}_{\mathsf{F},0}(\mathcal{A})]$.

Symmetric encryption. A symmetric encryption scheme SE specifies algorithms $\mathsf{SE.Sg}$, $\mathsf{SE.E}$, and $\mathsf{SE.D}$ (where the last of these is deterministic) and set $\mathsf{SE.M}$. State generation algorithm takes no input and outputs state σ which will be used as the initial encryption state σ^e and decryption state σ^d. Encryption algorithm $\mathsf{SE.E}$ takes as input σ^e and message $M \in \mathsf{SE.M}$. It outputs updated state σ^e and 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 $\mathsf{SE.D}$ takes as input σ^d and ciphertext C. It outputs updated state σ^d and $M \in \mathsf{SE.M} \cup \{\bot\}$. We write $\sigma \xleftarrow{\$} \mathsf{SE.Sg}$, $(\sigma^e, C) \xleftarrow{\$} \mathsf{SE.E}(\sigma^e, M)$, and $(\sigma^d, M) \leftarrow \mathsf{SE.D}(\sigma^d, C)$.

Correctness requires for all states $\sigma^e_0 = \sigma^d_0 \in [\mathsf{SE.Sg}]$ and all sequences of messages $\boldsymbol{M} \in (\mathsf{SE.M})^*$ that $\Pr[\forall i : \boldsymbol{M}_i = \boldsymbol{M}'_i] = 1$ where the probability is over the coins of encryption in the operations $(\sigma^e_i, \boldsymbol{C}_i) \xleftarrow{\$} \mathsf{SE.E}(\sigma^e_{i-1}, \boldsymbol{M}_i)$ and $(\sigma^d_i, \boldsymbol{M}'_i) \leftarrow \mathsf{SE.D}(\sigma^d_{i-1}, \boldsymbol{C}_i)$ for $i = 1, \ldots, |\boldsymbol{M}|$.

This non-standard syntax is used to simultaneously capture *stateful deterministic encryption* and *stateless probabilistic encryption*. For the first of these

$\mathsf{SE.E}$ is a deterministic algorithm. For the latter, σ^e and σ^d are equal to some key K which is never updated.

ENCRYPTION SECURITY. For security we will require that the output of encryption look like a random string. Consider the game $\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},b}(\mathcal{A})$ shown on the right side of Figure 2. 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 $\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{SE}}(\mathcal{A}) = \Pr[\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},1}(\mathcal{A})] - \Pr[\mathsf{G}^{\mathsf{indr}}_{\mathsf{SE},0}(\mathcal{A})]$.

3 The Switching Lemma

How hard is it for a memory-bounded distinguisher to tell apart a random function from a random permutation $[N] \to [N]$? It is easy to do so in a near-memory-less strategy with roughly $\sqrt{N}$ queries, where N is the domain size: The distinguisher, given access to an oracle $[N] \to [N]$, mounts a classical memory-less collision finding attack – if the attack succeeds, the distinguisher is highly certain it is interacting with a random function.

However, this attack requires querying the random function at the same point *twice*. It is not clear if a distinguisher which never repeats a query can still succeed with low memory and roughly $\sqrt{N}$ queries. We will show that it cannot. This boils down to bounding how well a memory-bounded can distinguish between a sequence of random values and a sequence of random values without repetition.

3.1 Streaming Indistinguishability

We consider a streaming setting, where a sequence of random variables

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

with range $[N]$ is given, one by one, to a (memory-bounded) distinguisher $\mathcal{A}$, which is otherwise computationally unbounded. The distinguisher will need to tell apart this setting from another one, where it is given $(Y_1, Y_2, \ldots, Y_q)$ instead. We are interested in its distinguishing advantage. This is a very natural setting, but we are not aware of this having been considered explicitly.

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 asks for the value $V_i \in \{X_i, Y_i\}$ based on which it updates its state to σ_i. We write for notational convenience $\mathcal{A}(i, \sigma_{i-1}, V_i) = \sigma_i$, noting that this mapping can be randomized. We denote in particular $\Sigma_0, \Sigma_1, \ldots, \Sigma_q$ the states during the execution with X^q and $\Gamma_0, \Gamma_1, \ldots, \Gamma_q$ the states during the execution with Y^q. Here $\Sigma_0 = \Gamma_0$ is some a priori fixed value. For the final state (Σ_q or Γ_q) $\mathcal{A}$ outputs a bit, which we denote by $\mathcal{A}(X^q)$ and $\mathcal{A}(Y^q)$, respectively, and we are interested in its advantage

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

It will sometime be convenient to think of this as an interaction between $\mathcal{A}$ and an oracle Samp which returns V_i's according to one of these distributions (written as $b \xleftarrow{\$} \mathcal{A}^{\textsc{Samp}}$).

We will use the following lemma below, for the case where the X_i's are individually uniformly distributed.

Lemma 2. *Let $X^q = X_1, \ldots, X_q$ be independent and uniformly distributed. Then for any $Y^q = Y_1, \ldots, Y_q$,*

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

Proof. Since the final output bit is Σ_q and Γ_q, respectively, we can always upper bound the advantage by the statistical distance of these states, i.e.,

$$\mathsf{Adv}^{\mathsf{dist}}_{X^q, Y^q}(\mathcal{A}) \leqslant \mathsf{SD}(\Sigma_q, \Gamma_q) = \mathsf{SD}(\Gamma_q, \Sigma_q) \ .$$

We will work in the regime of KL-divergence, and thus we also have

$$\mathsf{Adv}^{\mathsf{dist}}_{X^q, Y^q}(\mathcal{A}) \leqslant \frac{1}{\sqrt{2}} \sqrt{\mathsf{KL}(\Gamma_q \| \Sigma_q)} \ .$$

We note now that for all $i \in [q]$, by Lemma 1,

$$\mathsf{KL}(\Gamma_i \| \Sigma_i) = \mathsf{KL}(\mathcal{A}(i, \Gamma_{i-1}, Y_i) \| \mathcal{A}(i, \Sigma_{i-1}, X_i)) \leqslant \mathsf{KL}((\Gamma_{i-1}, Y_i) \| (\Sigma_{i-1}, X_i)) \ .$$

Write $P(s, x) = \Pr\left[(\Sigma_{i-1}, X_i) = (s, x)\right]$, $P(s) = \Pr\left[\Sigma_{i-1} = s\right]$ and $P(x|s) = \Pr\left[X_i = x \,\middle|\, \Sigma_{i-1} = s\right]$. Also define analogously $Q(s, x)$, $Q(s)$ and $Q(x|s)$ replacing (Σ_{i-1}, X_i) with (Γ_{i-1}, Y_i). Then,

$$\begin{aligned}
\mathsf{KL}((\Gamma_{i-1}, Y_i) \| (\Sigma_{i-1}, X_i)) &= \sum_{s,x} Q(s, x) \log \left(\frac{Q(s, x)}{P(s, x)} \right) \\
&= \sum_{s,x} Q(s, x) \log \left(\frac{Q(s)}{P(s)} \right) + \sum_{s,x} Q(s, x) \log \left(\frac{Q(x|s)}{P(x|s)} \right) \\
&= \mathsf{KL}(\Gamma_{i-1} \parallel \Sigma_{i-1}) + \log N - \sum_{s} Q(s) \log \left(\frac{1}{Q(x|s)} \right) \\
&= \mathsf{KL}(\Gamma_{i-1} \parallel \Sigma_{i-1}) + \log N - \mathsf{H}(Y_i \mid \Gamma_{i-1}) \ .
\end{aligned}$$

Therefore, $\mathsf{KL}(\Gamma_q \mid S_q) \leqslant \mathsf{KL}(\Gamma_0 \parallel S_0) + q \log N - \sum_{i=1}^{q} \mathsf{H}(Y_i \parallel \Gamma_{i-1})$, and the lemma follows since $\mathsf{KL}(\Gamma_0 \parallel S_0) = 0$. $\square$

3.2 Sampling with and without replacement

Consider the streaming indistinguishability of the following natural distributions:

- SAMPLING WITH REPLACEMENT. In the distribution $X^q = (X_1, X_2, \ldots, X_q)$ the X_i's are independent and uniformly distributed over $[N]$.
- SAMPLING WITHOUT REPLACEMENT. In the distribution $Y^q = (Y_1, \ldots, Y_q)$ the Y_i's are sampled uniformly *without* repetition from $[N]$ (thus $q \leqslant N$).

We want to upper bound the advantage in distinguishing these two streams for a memory-bounded distinguisher $\mathcal{A}$ which receives these values one by one. We are going to show a time-memory trade-off for any distinguisher $\mathcal{A}$, assuming a conjecture that we now state. We will discuss the conjecture (and *why* this requires a conjecture) later in Section 3.5.

A CONJECTURE ON HYPERGRAPHS. A *k-uniform simple hypergraph* (or henceforth, simply, a k-hypergraph) with N vertices and m edges is a collection $H = \{e_1, e_2, \ldots, e_m\}$ of *distinct* subsets $e_i \subseteq [N]$, each of size k. Conventional graphs correspond to the case $k = 2$. The *degree* $d_H(i)$ of a vertex $i \in [N]$ is

$$d_H(i) = |\{j \in [m] \,:\, i \in e_j\}| \ ,$$

i.e., the number of edges e_j containing i. By a double-counting argument we have $\sum_{i=1}^N d_H(i) = k \cdot m$. We will be interested in the following function of the degrees of a hypergraph,

$$D^2(H) = \sum_{i=1}^N d_H(i)^2 \ .$$

For example, if H is the complete k-hypergraph, i.e., it contains all $\binom{N}{k}$ possible edges, $d_H(i) = \binom{N-1}{k-1}$ for all $i \in [N]$, and thus $D^2(H) = N \cdot \binom{N-1}{k-1}^2$.

Let $\mathcal{H}_{N,k}(m)$ be the set of all k-hypergraphs with N vertices and m edges. We define in particular,

$$D^2_{N,k}(m) = \max_{H \in \mathcal{H}_{N,k}(m)} D^2(H) \ .$$

The behavior of $D^2_{N,2}(m)$ is fully characterized by a series of papers [14,10,20,1]. However, very little is known about $D^2_{N,k}(m)$ for general k. We will need the following conjecture.

Conjecture 1 (Main conjecture). Let $k > N/2$ and assume further that $m = \binom{A}{k}$ for some $A \geqslant k$. Then, the graph $H = \{e_1, ..., e_m\}$, where $e_1, \ldots, e_m$ are all size k subsets of $\{1, \ldots, A\}$, maximizes $D^2_{N,k}(m)$.

We refer the reader to Section 3.5 for an in-depth discussion of why we believe Conjecture 1 to be true, and why it is however hard to provide a full proof. We stress however that even weaker form of the conjecture (e.g., assuming that $D^2_{N,k}(m)$ is at most $(1 + 1/k)$ higher than the value achieved by the complete H) would not invalidate our bound below. Weakening even further would also simply result in a somewhat weaker bound.

Indistinguishability. We are going to now prove the following theorem.

Theorem 1. *Let N be given, $q < N/2$, and $20\log(e) \leqslant S \leqslant N/4$. Further, let X^q be sampled with replacement and Y^q be sampled without replacement from $[N]$. Then, if Conjecture 1 holds, for every S-bounded distinguisher $\mathcal{A}$, we have*

$$\mathsf{Adv}^{\mathsf{dist}}_{X^q,Y^q}(\mathcal{A}) \leqslant \sqrt{\frac{S \cdot q}{N}} \ .$$

Let $\mathcal{O}_{\mathrm{sl}}(q, S, N)$ denote the best possible advantage over all S-bounded adversaries. The above result tells us that $\mathcal{O}_{\mathrm{sl}} \in O(\sqrt{S \cdot q/N})$. For the sake of generality our results which use Theorem 1 are stated in terms of $\mathcal{O}_{\mathrm{sl}}$.

3.3 Proof of Theorem 1

We are going to use Lemma 2, and therefore we are going to be concerned solely with showing a lower bound on $\mathsf{H}(Y_i \parallel \Gamma_{i-1})$ for all $i \in [q]$. This involves in particular a random experiment where (1) $Y_1, \ldots, Y_i$ are sampled, and (2) the state Γ_{i-1} is going to be produced, as a function of $Y_1, \ldots, Y_{i-1}$ only (which however, also of course depend on Y_i by being distinct from it).

Intermediate experiment. We note that in the actual random experiment $\mathcal{A}$ has, when outputting Γ_{i-1}, information about $Y_1, \ldots, Y_{i-1}$ which is potentially incomplete, especially if Γ_{i-2} does not allow completely to remember $Y_1, \ldots, Y_{i-2}$, and so on. As a first simplification, we will remove this, and allow an adversary *full* information about $Y_1, \ldots, Y_{i-1}$ when attempting to produce a state Γ_{i-1} with the sole intent of making $\mathsf{H}(Y_i \mid \Gamma_{i-1})$ as small as possible. A second simplification is that, intuitively, the only information $Y_1, \ldots, Y_{i-1}$ give about Y_i is its range, i.e., the set of values Y_i can take.

In particular, for an adversary $\mathcal{B}$, consider the following experiment, producing variables (Y_i, Γ_{i-1}):

- Sample $\mathcal{Y} \xleftarrow{\$} \binom{[N]}{N-i+1}$
- Let $\Gamma_{i-1} \xleftarrow{\$} \mathcal{B}(\mathcal{Y})$
- $Y_i \xleftarrow{\$} \mathcal{Y}$
- Return (Y_i, Γ_{i-1})

The additional constraint here is that $|\Gamma_{i-1}| \leqslant S$. Define now $\mathsf{H}^i(\mathcal{B}) = \mathsf{H}(Y_i \mid \Gamma_{i-1})$. We will show the following.

Lemma 3. *For all i, and S-bounded adversary $\mathcal{A}$, there exists a* deterministic *$\mathcal{B}$ outputting at most S bits such that*

$$\mathsf{H}(Y_i \mid \Gamma_{i-1}) \geqslant \mathsf{H}^i(\mathcal{B}) \ ,$$

where $\mathsf{H}(Y_i \mid \Gamma_{i-1})$ is with respect to the original experiment.

Proof. We first build a randomized adversary $\mathcal{A}'$ which given $\mathcal{Y}$ first samples a random shuffling $Y_1, \ldots, Y_{i-1}$ of the $i-1$ elements *not* in $\mathcal{Y}$, and then runs $\mathcal{A}$ over $i-1$ rounds feeding $Y_1, \ldots, Y_{i-1}$ to it, to produce Γ_{i-1}, which is then output by $\mathcal{A}'$. Clearly, by construction, $\mathsf{H}(Y_i \mid \Gamma_{i-1}) = \mathsf{H}^i(\mathcal{B})$.

To make $\mathcal{B}$ deterministic, let R be the random coins used by $\mathcal{A}'$, and observe that

$$\mathsf{H}(Y_i \mid \Gamma_{i-1}) \geqslant \mathsf{H}(Y_i \mid \Gamma_{i-1}, R) = \mathop{\mathbf{E}}_{r \xleftarrow{\$} R} [\mathsf{H}(Y_i \mid \Gamma_{i-1}, R = r)] \ .$$

Define $\mathcal{B}$ by fixing the coins of $\mathcal{A}'$ to those minimizing $\mathsf{H}(Y_i \mid \Gamma_{i-1}, R = r)$. $\square$

Collision entropy and probabilities. We take an extra final step to simplify our lower bound, and its connection with Conjecture 1. Namely, we will lower bound

$$\mathsf{H}_2^i(\mathcal{B}) = \mathop{\mathsf{E}}_{\gamma \xleftarrow{\$} \Gamma_{i-1}} [\mathsf{H}_2(Y_i \mid \Gamma_{i-1} = \gamma)]$$

since clearly $\mathsf{H}^i(\mathcal{B}) \geqslant \mathsf{H}_2^i(\mathcal{B})$. Also define

$$\mathsf{Coll}^i(\mathcal{B}) = \mathop{\mathsf{E}}_{\gamma \xleftarrow{\$} \Gamma_{i-1}} \left[\sum_y \mathsf{Pr}\left[Y_i = y \,\middle|\, \Gamma_{i-1} = \gamma\right]^2 \right] \ .$$

We note here that by Jensen's inequality,

$$\mathsf{H}_2^i(\mathcal{B}) = \mathop{\mathsf{E}}_{\gamma \xleftarrow{\$} \Gamma_{i-1}} \left[-\log \left(\sum_y \mathsf{Pr}\left[Y_i = y \,\middle|\, \Gamma_{i-1} = \gamma\right]^2 \right) \right] \geqslant -\log \mathsf{Coll}^i(\mathcal{B}) \ ,$$

because $x \mapsto -\log(x)$ is a convex function. Therefore, the rest of the section will be devoted to proving an upper bound for $\mathsf{Coll}^i(\mathcal{B})$. Specifically, we show:

Lemma 4. *For all adversaries $\mathcal{B}$ outputting at most S bits, if Conjecture 1 is true,*

$$\mathsf{Coll}^i(\mathcal{B}) \leqslant \left(1 + \frac{2}{N}\right) \cdot \frac{1}{N - S} \ .$$

Before we turn to a proof, let us see how this implies the desired result. First off, it immediately implies by the above

$$\begin{aligned}
\mathsf{H}(Y_i \mid \Gamma_{i-1}) &\geqslant -\log \mathsf{Coll}^i(\mathcal{B}) \\
&\geqslant -\log\left(1 + \frac{2}{N}\right) + \log(N - S) \\
&= -\log\left(1 + \frac{2}{N}\right) + \log(N) + \log\left(1 - \frac{S}{N}\right) \ .
\end{aligned}$$

Now note that $\log(1+x) \leqslant \log(e^x) = x \log(e)$. On the other hand, using the fact that $x = S/N \leqslant 0.25$, we have

$$\log(1 - x) = \frac{1}{\ln 2} \ln(1 - x) \geqslant \frac{1}{\ln 2} \left(-x - x^2/2 - x^3/2\right) \geqslant \frac{-21x}{16 \ln 2} \geqslant -1.9x$$

Plugging in gives,

$$\sum_{i=1}^{q} \mathsf{H}(Y_i \mid \Gamma_{i-1}) \geqslant q\left(-\frac{2\log(e)}{N} + \log(N) - \frac{1.9S}{N}\right) .$$

Then using Lemma 2 we can complete the proof via

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{dist}}_{X^q,Y^q}(\mathcal{A}) &\leqslant \frac{1}{\sqrt{2}}\sqrt{q\log N - \sum_{i=1}^{q}\mathsf{H}(Y_i \mid \Gamma_i)} \\
&\leqslant \frac{1}{\sqrt{2}}\sqrt{q\left(\frac{2\log(e)}{N} + \frac{1.9S}{N}\right)} \\
&\leqslant \frac{1}{\sqrt{2}}\sqrt{q\left(\frac{0.1S}{N} + \frac{1.9S}{N}\right)} = \sqrt{\frac{S\cdot q}{N}} .
\end{aligned}$$

Proof of Lemma 4. We first introduce some more notation. For a k-hypergraph $H = \{e_1, \ldots, e_m\}$ with vertex set $[N]$ where $k := N - i + 1$, consider the distribution p_H which samples a $y \in [N]$ by first picking a random edge e_i, and then letting y be a random element of the set. In particular, $p_H(y) = d_H(y)/m \cdot k$. We also define

$$\mathsf{Coll}(H) = \sum_y p_H(y)^2 = \frac{1}{m^2k^2}D^2(H) .$$

Also, let $\mathsf{Coll}_{N,k}(m) = \max_{H\in\mathcal{H}_{N,k}(m)} \mathsf{Coll}(H)$.

Note now that $\mathcal{B}$ assigns sets of size k to every S-bit output γ. For a given γ, we can think of the sets assigned to it as a k-hypergraph, which we denote $\mathcal{B}^{-1}(\gamma)$. Letting m_γ denote the number of edges in $\mathcal{B}^{-1}(\gamma)$ (and thus $\sum_\gamma m_\gamma = \binom{N}{k}$), we have

$$\mathsf{Coll}(\mathcal{B}) = \frac{1}{\binom{N}{k}} \sum_{\gamma\in\{0,1\}^S} m_\gamma \cdot \mathsf{Coll}(\mathcal{B}^{-1}(\gamma)) \leqslant \frac{1}{\binom{N}{k}} \sum_{\gamma\in\{0,1\}^S} m_\gamma \cdot \mathsf{Coll}_{N,k}(m_\gamma) . \quad (1)$$

We are going to now maximize the right-hand-side of the above inequality over all sets $\{m_\gamma\}_{\gamma\in\{0,1\}^S}$, where $\sum_\gamma m_\gamma = \binom{N}{k}$, using Conjecture 1.[4] We need the following helping lemma, that $\mathsf{Coll}_{N,k}(m_\gamma)$ is a non-increasing function. Its proof is deferred to the full version of this paper [19].

Lemma 5. *For all $m \geqslant 1$, $\mathsf{Coll}_{N,k}(m+1) \leqslant \mathsf{Coll}_{N,k}(m)$.*

Unfortunately, the function $\mathsf{Coll}_{N,k}(m)$ is not "smooth", due to its discrete nature, making our maximization of the RHS of (1) difficult. We will now replace it with a continous version without too much loss. Concretely, we define

$$A_{N,k}(m) = \frac{1}{\alpha} ,$$

[4] Note that applying this conjecture requires $k > N/2$ which holds because $k = N - i + 1 \geqslant N - q + 1 > N - N/2 + 1$.

where $\alpha \in [k, N]$ is the (unique) real number such that

$$\binom{\alpha}{k} = \frac{\alpha(\alpha-1)\cdots(\alpha-k+1)}{k!} = m \ .$$

We can now use the following lemma.

Lemma 6. *Assume Conjecture 1. For all $m \in \{1, 2, \ldots, \binom{N}{k}\}$, we have*

$$\mathsf{Coll}_{N,k}(m) \leqslant \left(1 + \frac{1}{k}\right) \cdot A_{N,k}(m) \ .$$

Proof. Pick m, and let $m_0 \leqslant m \leqslant m_1$ such that $m_0 = \binom{A}{k}$ and $m_1 = \binom{A+1}{k}$ for a natural number $A \geqslant k$. Then, $A_{N,k}(m) = \frac{1}{\alpha}$ for some $\alpha \in [A, A+1]$, and using Lemma 5 and Conjecture 1,

$$\mathsf{Coll}_{N,k}(m) \leqslant \mathsf{Coll}_{N,k}(m_0) = \frac{1}{A} = \frac{\alpha}{A} A_{N,k}(m) \leqslant \frac{1+A}{A} A_{N,k}(m) \ .$$

The claim follows, because $\frac{1+A}{A} \leqslant 1 + \frac{1}{k}$. ◻

Therefore, we can now adapt this to (1) as

$$\begin{aligned} \mathsf{Coll}(\mathcal{B}) &\leqslant \left(1 + \frac{1}{k}\right) \frac{1}{\binom{N}{k}} \sum_{\gamma \in \{0,1\}^S} m_\gamma \cdot A_{N,k}(m_\gamma) \\ &= \left(1 + \frac{1}{k}\right) \frac{1}{\binom{N}{k}} \sum_{\gamma \in \{0,1\}^S} B_{N,k}(m_\gamma) \ , \end{aligned} \tag{2}$$

where $B_{N,k}(m) = m \cdot A_{N,k}(m)$. To conclude the proof, we use the following two lemmas, whose proofs are deferred to the full version of this paper [19].

Lemma 7. *The function $B_{N,k}(m)$ is concave.*

Lemma 8. *For $N/2 \leqslant k \leqslant N - S$, we have $\binom{N}{k}/2^S \geqslant \binom{N-S}{k}$.*

Lemma 7 can now be applied to (2) to yield

$$\begin{aligned} \mathsf{Coll}(\mathcal{B}) &\leqslant \left(1 + \frac{1}{k}\right) \frac{2^S}{\binom{N}{k}} \frac{1}{2^S} \sum_{\gamma \in \{0,1\}^S} B_{N,k}(m_\gamma) \\ &\leqslant \left(1 + \frac{1}{k}\right) \frac{2^S}{\binom{N}{k}} B_{N,k}\left(\frac{1}{2^S} \sum_{\gamma \in \{0,1\}^S} m_\gamma\right) \\ &= \left(1 + \frac{1}{k}\right) \frac{2^S}{\binom{N}{k}} B_{N,k}\left(\binom{N}{k}/2^S\right) \\ &= \left(1 + \frac{1}{k}\right) \cdot A_{N,k}\left(\binom{N}{k}/2^S\right) \\ &\leqslant \left(1 + \frac{1}{k}\right) \cdot A_{N,k}\left(\binom{N-S}{k}\right) = \left(1 + \frac{1}{k}\right) \frac{1}{N-S} \ , \end{aligned} \tag{3}$$

where for the last inequality we have used Lemma 8 and the fact that $A_{N,k}(\cdot)$ is a non-increasing function.

3.4 Application: The Switching Lemma and Counter-mode encryption

THE SWITCHING LEMMA. A classic result in cryptography is the *switching lemma* which says roughly that for any blockcipher F and adversary $\mathcal{A}$ making at most q oracle queries, $\left|\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}) - \mathsf{Adv}^{\mathsf{prp}}_{\mathsf{F}}(\mathcal{A})\right| < q^2/N$ where $N = |\mathsf{F.Dom}|$. The standard proof works by bounding the ability of $\mathcal{A}$ to distinguish a random function from a random permutation by analyzing the probability that the output of a random function repeats. When $\mathcal{A}$ does not repeat its oracle queries we can reduce this to the streaming problem we just analyzed this.

Lemma 9. *Let* F *be a blockcipher with* $\mathsf{F.Dom} = [N]$*. Let* $\mathcal{A}$ *be an* S*-bounded adversary which makes at most* q non-repeating *queries to its oracle. Then*

$$|\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}) - \mathsf{Adv}^{\mathsf{prp}}_{\mathsf{F}}(\mathcal{A})| \leqslant \mathcal{O}_{\mathrm{sl}}(q, S, N) \; .$$

If Conjecture 1 holds, then we can in turn bound $\mathcal{O}_{\mathrm{sl}}(q, S, N)$ by $\sqrt{S \cdot q/N}$ using Theorem 1. This would make the bound (and others in the section) essentially tight. If an attacker stores S outputs from its oracle, we expect it to see one of these outputs again from a random function after $T \approx N/S$ queries. For a random permutation such a repeat is impossible. In the full version of this paper [19] we provide the (simple) analysis for this attack.

Proof. Without loss of generality, assume that $\mathsf{Adv}^{\mathsf{dist}}_{X^q,Y^q}(\mathcal{A})$ is positive. We claim that $\Pr[\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},0}(\mathcal{A})] = \Pr[\mathcal{A}(X^q) \Rightarrow 1]$ and $\Pr[\mathsf{G}^{\mathsf{prp}}_{\mathsf{F},0}(\mathcal{A})] = \Pr[\mathcal{A}(Y^q) \Rightarrow 1]$. Then the following calculation establishes the result.

$$\begin{aligned}
|\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}) - \mathsf{Adv}^{\mathsf{prp}}_{\mathsf{F}}(\mathcal{A})| &= |\Pr[\mathsf{G}^{\mathsf{prp}}_{\mathsf{F},0}(\mathcal{A})] - \Pr[\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},0}(\mathcal{A})]| \\
&= |\Pr[\mathcal{A}(Y^q) \Rightarrow 1] - \Pr[\mathcal{A}(X^q) \Rightarrow 1]| \\
&= \mathsf{Adv}^{\mathsf{dist}}_{X^q,Y^q}(\mathcal{A}) \\
&\leqslant \mathcal{O}_{\mathrm{sl}}(q, S, N) \; .
\end{aligned}$$

The first equality used that games $\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},1}(\mathcal{A})$ and $\mathsf{G}^{\mathsf{prp}}_{\mathsf{F}.1}(\mathcal{A})$ are identical. □

COUNTER-MODE ENCRYPTION. Let F be a family of functions with $\mathsf{F.Dom} = [N]$ for some $N \in \mathbb{N}$ and $\mathsf{F.Rng} = \{0,1\}^{\mathsf{F.ol}}$ for some $\mathsf{F.ol} \in \mathbb{N}$. One classic example of an encryption mode constructed using F is *stateful counter-mode*. Formally this is the encryption scheme $\mathsf{CTR[F]}$ with $\mathsf{CTR[F].M} = (\{0,1\}^{\mathsf{F.ol}})^*$ and algorithms defined as shown below.

$\underline{\mathsf{CTR[F].Sg}}$	$\underline{\mathsf{CTR[F].E}(\sigma^e, M)}$	$\underline{\mathsf{CTR[F].D}(\sigma^d, C)}$
$K \xleftarrow{\$} \mathsf{F.K}$	$(i, K) \leftarrow \sigma^e$	$(i, K) \leftarrow \sigma^d$
Return $(0, K)$	For $j = 0, \ldots, \lvert M\rvert_{\mathsf{F.ol}}$	For $j = 0, \ldots, \lvert C\rvert_{\mathsf{F.ol}}$
	$\quad C_j \leftarrow M_j \oplus \mathsf{F.Ev}(K, i+j)$	$\quad M_j \leftarrow C_j \oplus \mathsf{F.Ev}(K, i+j)$
	$i \leftarrow i + \lvert M\rvert_{\mathsf{F.ol}}$	$i \leftarrow i + \lvert C\rvert_{\mathsf{F.ol}}$
	Return $((i, K), C)$	Return $((i, K), M)$

Adversary $\mathcal{A}_{\mathsf{prf}}^{\textsc{Ror}}$	$\textsc{SimEnc}(M)$
$i \leftarrow 0$	$C \leftarrow M \oplus \textsc{Ror}(i)$
$b' \xleftarrow{\$} \mathcal{A}^{\textsc{SimEnc}}$	$i \leftarrow i+1$
Return b'	Return C

Fig. 3. Adversary for Theorem 2.

Here addition is mod N. It is trivial to show that if F is a good PRF then, $\mathsf{CTR[F]}$ is a secure encryption scheme. Consider the following theorem. For simplicity we focus on the case that the attacker queries only 1 block messages.

Theorem 2. *Let* F *be given with* $\mathsf{F.Dom} = [N]$ *and* $\mathsf{F.Rng} = \{0,1\}^{\mathsf{F.ol}}$. *Let* $\mathcal{A}$ *be an adversary making at most* $q < N$ *queries to its* $\textsc{Enc}$ *oracle where each is* $\mathsf{F.ol}$ *bits long. Then we can build an adversary* $\mathcal{A}_{\mathsf{prf}}$ *(Fig. 3) such that*

$$\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{CTR[F]}}(\mathcal{A}) = \mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prf}}) \ .$$

Adversary $\mathcal{A}_{\mathsf{prf}}$ *is roughly as efficient as* $\mathcal{A}$.

Proof. Let $\mathcal{A}_{\mathsf{prf}}$ be the adversary shown in Fig. 3. It uses its $\textsc{Ror}$ oracle to simulate the view of $\mathcal{A}$. We claim that $\Pr[\mathsf{G}^{\mathsf{indr}}_{\mathsf{CTR[F]},1}(\mathcal{A})] = \Pr[\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},1}(\mathcal{A})]$ and $\Pr[\mathsf{G}^{\mathsf{indr}}_{\mathsf{CTR[F]},0}(\mathcal{A})] = \Pr[\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},0}(\mathcal{A})]$ from which the stated advantage relationship follows. The former equality holds because in both $\mathcal{A}$ is seeing $\mathsf{CTR[F]}$ encryptions of M. For the latter equality note that the total block-length of all of $\mathcal{A}$'s queries is less than N so the input to the random function will never repeat. Consequently each value returned by $\textsc{Ror}$ in $\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},0}(\mathcal{A})$ (and thus each $C_j = M_j \oplus \textsc{Ror}(i+j)$) is a fresh random string. This is identical to the distribution on C returned to $\mathcal{A}$ in $\mathsf{G}^{\mathsf{indr}}_{\mathsf{CTR[F]},0}(\mathcal{A})$.

The efficiency of $\mathcal{A}_{\mathsf{prf}}$ can be verified by examining its pseudocode. $\square$

Suppose F is a blockcipher (where we identfy $[N]$ with $\{0,1\}^{\mathsf{F.ol}}$ in the obvious way). If $q \in \Omega(\sqrt{N})$, then we cannot generically hope that $\mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prf}})$ is small because an attacker with unbounded state can remember the outputs of F for every query it made and check if they ever repeated. However, if S is $o(\sqrt{N})$ then we can still meaningfully hope for security because $\mathcal{A}_{\mathsf{prf}}$ cannot remember ever query it made. In particular, by combining Thm. 2 and Lemma 9 we obtain the following corollary.

Corollary 1. *Let* F *be a blockcipher with* $\mathsf{F.Rng} = \{0,1\}^{\mathsf{F.ol}}$. *Let* $\mathcal{A}$ *be an* S*-bounded adversary making at most* $q \leqslant 2^{\mathsf{F.ol}}$ *queries to its* $\textsc{Enc}$ *oracle each of which are* $\mathsf{F.ol}$ *bits long. Then we can build an adversary* $\mathcal{A}_{\mathsf{prf}}$ *(Fig. 3) such that*

$$\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{CTR[F]}}(\mathcal{A}) \leqslant \mathsf{Adv}^{\mathsf{prp}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prf}}) + \mathcal{O}_{\mathrm{sl}}(q, S, 2^{\mathsf{F.ol}}) \ .$$

Adversary $\mathcal{A}_{\mathsf{prf}}$ *is roughly as efficient as* $\mathcal{A}$.

Proving this requires only observing that $\mathcal{A}_{\mathsf{prf}}$ is S-bounded. Examining the code of $\mathcal{A}_{\mathsf{prf}}$ it may seem like it needs to remember the counter i and M in addition to the state of $\mathcal{A}$. However, as per the computation model in Section 2.1, the stage number is given to an adversary during each stage and the i-th message M_i can be deterministically recomputed from $\mathcal{A}$'s state σ_{i-1}.

Output-feedback mode encryption. In the full version of this paper [19] we apply our streaming results to analyze the security of stateful output-feedback mode. This mode starts with $Y_0 = 0^{\mathsf{F.ol}}$ and the encrypts each M_i via $Y_i \leftarrow \mathsf{F.Ev}(K, Y_{i-1})$; $C_i \leftarrow M_i \oplus Y_i$ where F is a blockcipher. The analysis of the mode is more involved than the $\mathsf{CTR\$}$ analysis because we cannot a priori assume that the inputs to F will not repeat.

The crux of the proofs lies in considering the streaming problem of distinguishing 1, $F(1)$, $F(F(1))$, … from random where F is a random permutation $[N] \to [N]$. This is exactly what arises from the standard reduction to replace the PRF with a truly random function. In analyzing this streaming problem we first bound the statistical distance between the stated distribution and sampling without replacement. This gives a $O(q/N)$ term corresponding to the probability that 1 is chosen as the output of F for any of first q samples in the distribution. Having done this we can now simply apply a bound on the streaming problem we have been studying in this section. Putting everything together, the reduction from security of the encryption scheme to this new streaming problem is straightforward and gives a bound $\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{OFB[F]}}(\mathcal{A}) = \mathsf{Adv}^{\mathsf{prp}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prp}}) + \mathcal{O}_{\mathrm{sl}}(q, S, 2^{\mathsf{F.ol}}) + 4q/N$.

Surprisingly, this result *cannot* hold for output-feedback mode with a PRF instead of a PRP. In the full version of this paper [19] we note a low memory attack that with high success probability when the number of encrypted blocks is $\Omega(\sqrt{N})$. The critical difference allowing this attack is that random functions have much shorter cycle lengths than random permutations. The importance of cycle lengths for OFB was first noted by Davies and Parkin [13].

Nonce-based encryption. A standard way of constructing nonce-based encryption from a randomized encryption scheme is to apply a PRF to the nonce to obtain coins for the underlying encryption scheme. Because nonce repetitions are disallowed in the most basic security definitions for nonce-based encryption we can use Lemma 9 to replace the PRF with a PRP. The proof of this is straightforward and we omit a formalization.

3.5 Validity of Conjecture 1

We now discuss conjecture 1. First off, we point out that the problem is well understood for the case of graphs, that correspond to $k = 2$.

Additionally, note that the conjecture is not true for all k. For example, take $k = 2$, $m = \binom{4}{2} = 6$ and $N \geqslant 7$. The complete graph over 4 vertices gives $D^2(K_4) = 4 \times 9 = 36$. Yet the star S_6 with edges $\{1,2\}, \{1,3\}, \ldots, \{1,7\}$ has $D^2(S_6) = 6^2 + 6 \times 1 = 42$. In fact, one can show that S_6 is optimal (see below).

The case $k > N/2$. However, this is different for $k > N/2$, and we briefly explain the intuition, by giving an equivalent formulation of our conjecture.

The first observation here is that for any k-hypergraph $H = \{e_1, \ldots, e_m\}$, we can define its complement as the $(N-k)$-hypergraph $H' = \{e'_1, \ldots, e'_m\}$, where $e'_i = [N] \backslash e_i$. Now, note that

$$\begin{aligned} D^2(H) &= \sum_{i=1}^{N} d_H(i)^2 = \sum_{i=1}^{N} (m - d_{H'}(i))^2 \\ &= N \cdot m^2 - 2m \cdot \sum_{i=1}^{N} d_{H'}(i) + \sum_{i=1}^{N} d_{H'}(i)^2 \\ &= N \cdot m^2 - 2m^2(N-k) + D^2(H') \ . \end{aligned}$$

This in particular implies directly the following: H maximizes $D^2(H)$ over k-hypergraphs with m edges iff H' maximizes $D^2(H')$ over $(N-k)$-hypergraphs with m edges.

In general, if $m = \binom{A}{k}$ for $N/2 < k \leqslant A \leqslant N$, then our conjecture says that the complete k-hypergraph over $[A]$, denoted $K_{A,k}$, maximizes $D^2(H)$. We note that the complement of $K_{A,k}$ is (isomorphic to) $S_{N,N-A,N-k}$, where $S_{N,R,k'}$ for $k' > R$ is the k'-hypergraph with edges

$$\{1, \ldots, R\} \cup e \ ,$$

and e is any subset of size $k' - R$ of $\{R+1, \ldots, N\}$. Our conjecture is then equivalent to the statement that for any $k' < N/2$ and $m = \binom{A}{N-k'}$, the graph $H = S_{N,R,k'}$ for $R = N - A$ maximizes $D^2(H)$.

Example 1. The conjecture is easily seen to be true for $k = N-2$, and we are given $m = \binom{N-1}{N-2}$ edges (this is the only non-trivial m). Then, $k' = 2$, and thus $S_{N,N-A,N-k} = S_{N,1,2} = S_N$, the graph which contains exactly all edges $\{i, N\}$ for $i \in [N-1]$.

Now, we can see that $H = S_N$ maximizes $D^2(H)$. This is because for any k'-hypergraph $H = (e_1, \ldots, e_m)$, let $\mathbf{v}_1, \ldots, \mathbf{v}_m \in \{0,1\}^N$ be the characteristic vectors of the edges, then

$$\begin{aligned} D^2(H) &= \left(\sum_{i=1}^{m} \mathbf{v}_i \right)^T \left(\sum_{i=1}^{m} \mathbf{v}_i \right) \\ &= \sum_{i=1}^{m} \mathbf{v}_i^T \mathbf{v}_i + 2 \sum_{i,j} \mathbf{v}_i^T \mathbf{v}_j \\ &= m \cdot k' + 2 \sum_{i,j} |e_i \cap e_j| \ . \end{aligned}$$

Clearly, for edges of size $k' = 2$, $|e_i \cap e_j|$ is at most 1, and S_N has the property that it is *exactly* one for any $i \neq j$.

The above example, showing the optimality of one simple special case, also shows our intuition. Namely, to maximize $m \cdot k' + 2 \sum_{i,j} |e_i \cap e_j|$, we make every pair of vertices share the highest number of possible vertices, i.e., $N - A$. The number of edges then exactly corresponds to the completion of all edges consisting of all subsets of size A of the remaining vertices.

Dual graph. We can repeat an analogous analysis of the dual graph of $H = \{e_1, \ldots, e_m\}$. We define this to be the k-hypergraph $\overline{H} = \binom{[N]}{k} \setminus H$. Now, note that

$$\begin{aligned} D^2(H) = \sum_{i=1}^{N} d_H(i)^2 &= \sum_{i=1}^{N} \left(\binom{N}{k} - d_{H'}(i) \right)^2 \\ &= N \cdot \binom{N}{k}^2 - 2\binom{N}{k}^2 (N-k) + D^2(H') \ . \end{aligned}$$

This implies that H maximizes $D^2(H)$ over k-hypergraphs with m edges iff H' maximizes $D^2(H')$ over k-hypergraphs with $\binom{N}{k} - m$ edges.

The complement of a k-hypergraph $K_{A,k}$ is isomorphic to $Z_{N,N-A,k}$, where $Z_{N,R,k}$ is the k-hypergraph with all edges $e \in \binom{[N]}{k}$ such that

$$\{1, \ldots, R\} \cap e \neq \varnothing \ .$$

Our conjecture is then equivalent to the statement that for any $k > N/2$ and $m = \binom{A}{k}$, the graph $H = Z_{N,R,k}$ for $R = N - A$ maximizes $D^2(H)$. Note when $k = 2$, the only S graphs are isomorphic to $S_{N,1,2} = Z_{N,1,2}$. Furthermore, when $k = 2$ for an appropriate generalization of complete graphs and Z graphs (covering when they do not "fits" perfectly for a given m) $D^2(H)$ is *always* maximized by a complete or Z graph.

Complete, S, and Z graphs are very natural ways to try to "pack" a hypergraph. Complete graphs create a uniform packing over a subset of the nodes with no overflow. Both S and Z graphs create very biased packings by making a small subset of the nodes have particularly high degree at the expense of a long tail of nodes that have low, but non-zero degree.

Why proving it is hard? One reason why proving the conjecture is hard is that we are maximizing a function over degree sequences $(d_1, \ldots, d_N)$ of hypergraphs. The structure of this set is however not well understood, even when dropping the restriction that we must have exactly m edges.

4 Randomized Encryption

The general streaming setting introduced in Section 3.1 can be used to derive time-memory tradeoff bounds for other encryption schemes by considering other distributions for X^q and Y^q. In this section we study randomized stateless encryption schemes (the only state is an unchanging secret key K). Our main positive result is for randomized counter-mode (CTR\$) with a good PRF. Towards this we start by (in Section 4.1) specifying the necessary streaming distribution for analyzing CTR\$. Analyzing this requires different techniques than those used in Section 3.3 and is done *unconditionally* (i.e. we do not rely on Conjecture 1).

Note that, unlike in the case of stateful counter-mode, security with a PRF is not trivial because the input to the function may repeat across different encryption queries. We show a $O(\sqrt{Spq/N})$ bound on the adversary's advantage where p is the length of the messages encrypted and q is the number of messages. Note that the running time of an adversary, T, upper bounds $p \cdot q$.

Beyond this we show a generic "switching lemma" between two notions of weak PRF security. In the first an adversary tries to distinguish between $(R, \mathsf{F.Ev}(K, R))$ and $(R, F(R))$ for randomly sampled R and F a random function $[N] \to [N]$. In the other notion, the latter distribution is replaced with (R, Y) where Y is chosen at random. The latter of these is more useful for security, but the former is more plausible achieved with good bounds. We show that there can be at most an $O(\sqrt{ST/N})$ difference between an adversary's advantage in these two games. As an example application of this result we note this can be used to provide a time-memory tradeoff for the INDR security of the Encrypt-then-PRF generic composition.

All of these bounds are essentially tight. If an attacker stores S input-output examples for F, we expect it to see one of these inputs again (allowing it to trivially distinguish from random) after $T \approx N/S$ queries.

4.1 Streaming distributions for $\mathsf{CTR\$}$

Consider the streaming indistinguishability of the following two distributions.

- $\underline{\textsc{Rand}[N, M, p, q]}$. The distribution $X^q = (X_1, X_2, \ldots, X_q)$ is such that the X_i's are independent and uniformly distributed over $[N] \times [M]^p$.
- $\underline{\mathrm{CTR\$}[N, \mathcal{F}, p, q]}$. For the distribution $Y^q = (Y_1, \ldots, Y_q)$ first a function F is sampled at random from $\mathcal{F}$. Then $Y_i = (R_i, F(R_i + 1), \ldots, F(R_i + p))$ where R_i's are are independent and uniformly distributed over $[N]$ and addition is modulo N.

To analyze $\mathsf{CTR\$}$ with a good PRF we will let $\mathcal{F} = \mathsf{Fcs}(N, M)$. Security with a good PRP could be modeled by letting $N = M$ and $\mathcal{F} = \mathsf{Perm}(N)$.

Indistinguishability. We are going to now prove the following theorem.

Theorem 3. *Let N, M, p, q, and S be given such that $p|N$. Furthermore, let $X^q = \textsc{Rand}[N, M, p, q]$ and $Y^q = \mathrm{CTR\$}[N, \mathsf{Fcs}(N, M), p, q]$. Then for every S-bounded distinguisher $\mathcal{A}$, we have*

$$\mathsf{Adv}^{\mathsf{dist}}_{X^q, Y^q}(\mathcal{A}) \leqslant \frac{1}{\sqrt{2}} \sqrt{\frac{S \cdot p \cdot q}{N}} .$$

Note that unlike Theorem 1 we prove this result uncategorically, without requiring any conjectures.

For notational convenience we use bold-face to indicate vectors obtained by adding 1 through p to some value. For example, if $R \in [N]$ we will let $\boldsymbol{R} = (R + 1, \ldots, R + p)$. Further, we let $F(\boldsymbol{R}) = (F(R + 1), \ldots, F(R + p))$.

In the proof we will use the chain rule which says $H(X, Y) = H(X|Y) + H(Y)$. We also use that $H(X, Y \mid Z) \leqslant H(X \mid Z) + H(Y \mid Z)$ and $H(X) \leqslant \log \mathcal{X}$ where $\mathcal{X}$ is the support of X with equality when X is uniformly distributed over $\mathcal{X}$. These are standard facts about entropy.

4.2 Proof of Theorem 3

Associating the set $[N] \times [M]^p$ with $[N \cdot M^p]$ we can use Lemma 2 to obtain a bound of,

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

Therefore we are going to be concerned solely with showing a lower bound on $\mathsf{H}(Y_i \mid \Gamma_i)$ for all $i \in [q]$. Recall that Y_i is the tuple $(R_i, F(\boldsymbol{R}_i))$. The chain rule gives that $\mathsf{H}(Y_i \mid \Gamma_i) = H(F(\boldsymbol{R}_i) \mid R_i, \Gamma_i) + H(R_i \mid \Gamma_i)$.

Note that R_i is independent of Γ_i and uniformly sampled from $[N]$ so $H(R_i \mid \Gamma_i) = \log N$. Conditioning over all possible values of R_i gives

$$H(F(\boldsymbol{R}_i) \mid R_i, \Gamma_i) = N^{-1} \cdot \sum_{r \in [N]} H(F(\boldsymbol{r}) \mid \Gamma_{i-1})\ .$$

Observe that because p divides N the vectors $\boldsymbol{r}$ can be divided into p different partitions of $[N]$. That is for every $j \in [p]$, $\bigsqcup_{k \in [N/p]}\{j+kp+1, \ldots, j+kp+p\} = [N]$. This observation allows us to continue our calculations as follows,

$$\begin{aligned}
H(F(\boldsymbol{R}_i) \mid R_i, \Gamma_i) &= N^{-1} \cdot \sum_{j \in [p]} \sum_{k \in [N/p]} H(F(\boldsymbol{j} + \boldsymbol{kp}) \mid \Gamma_{i-1}) \\
&\geqslant N^{-1} \cdot p \cdot H(F \mid \Gamma_{i-1}) \\
&\geqslant N^{-1} \cdot p \cdot (H(F) - H(\Gamma_{i-1})) \\
&\geqslant N^{-1} \cdot p \cdot (N\log M - S)\ . \\
\text{Thence } \sum_{i=1}^{q} \mathsf{H}(Y_i \mid \Gamma_i) &= \sum_{i=1}^{q} H(F(\boldsymbol{R}_i) \mid R_i, \Gamma_i) + H(R_i \mid \Gamma_i) \\
&\geqslant \sum_{i=1}^{q} N^{-1} \cdot p \cdot (N\log M - S) + \log N \\
&= q\log(N \cdot M^p) - Spq/N\ ,
\end{aligned}$$

from which the result follows. □

4.3 Application: CTR\$ with a PRF and Weak PRFs

RANDOMIZED COUNTER-MODE. We can use Theorem 3 to prove a security result for randomized counter-mode encryption. Let F be a family of functions with $\mathsf{F.Dom} = [N]$ and $\mathsf{F.Rng} = \{0,1\}^{\mathsf{F.ol}}$. Then randomized counter-mode with F is the encryption scheme $\mathsf{CTR\$[F]}$ with state generation algorithm $\mathsf{CTR\$[F].Sg} = \mathsf{F.K}$, message space $\mathsf{CTR\$[F].M} = (\{0,1\}^{\mathsf{F.ol}})^*$, and encryption/decryption algorithms defined as shown below.

Adversary $\mathcal{A}_{\mathsf{prf}}^{\textsc{Ror}}$	Distinguisher $\mathcal{A}_{\mathsf{dist}}^{\textsc{Samp}}$
$b' \xleftarrow{\$} \mathcal{A}^{\textsc{SimEnc}}$	$b' \xleftarrow{\$} \mathcal{A}^{\textsc{SimEnc}}$
Return b'	Return b'
$\underline{\textsc{SimEnc}(M)}$	$\underline{\textsc{SimEnc}(M)}$
$R \xleftarrow{\$} [N]$	$(R, V_1, \ldots, V_p) \leftarrow \textsc{Samp}$
For $i = 1, \ldots, \|M\|_{\mathsf{F.ol}}$ do	For $i = 1, \ldots, \|M\|_{\mathsf{F.ol}}$ do
$\quad C_i \leftarrow M_i \oplus \textsc{Ror}(R + i)$	$\quad C_i \leftarrow M_i \oplus V_i$
Return (R, C)	Return (R, C)

Fig. 4. Adversary for Theorem 4.

$\underline{\mathsf{CTR\$[F].E}(K, M)}$	$\underline{\mathsf{CTR\$[F].D}(K, (R, C))}$
$R \xleftarrow{\$} [N]$	For $i = 1, \ldots, \|C\|_{\mathsf{F.ol}}$
For $i = 1, \ldots, \|M\|_{\mathsf{F.ol}}$	$\quad M_i \leftarrow C_i \oplus \mathsf{F.Ev}(K, R + i)$
$\quad C_i \leftarrow M_i \oplus \mathsf{F.Ev}(K, R + i)$	Return (K, M)
Return $(K, (R, C))$	

Here $R + i$ is addition mod N. The standard security theorem for $\mathsf{CTR\$[F]}$ tells us (roughly) that given an adversary $\mathcal{A}$ making q oracle queries we can construct a PRF adversary $\mathcal{A}_{\mathsf{prf}}$ such that $\mathsf{Adv}_{\mathsf{SE}}^{\mathsf{indr}}(\mathcal{A}) \leqslant \mathsf{Adv}_{\mathsf{F}}^{\mathsf{prf}}(\mathcal{A}_{\mathsf{prf}}) + p^2q^2/N$. Below is our theorem which takes space into account to provide a better bound when the amount of space used is much less than pq.

Theorem 4. *Let* F *be a family of functions with* $\mathsf{F.Dom} = [N]$ *and* $\mathsf{F.Rng} = \{0,1\}^{\mathsf{F.ol}}$. *Let* $\mathcal{A}$ *be an* S*-bounded adversary making at most* q *queries with lengths at most* $p \cdot \mathsf{F.ol}$ *bits to its oracle. Assume* $p|N$. *Then we can build an adversary* $\mathcal{A}_{\mathsf{prf}}$ *(Fig. 4) such that*

$$\mathsf{Adv}_{\mathsf{CTR\$[F]}}^{\mathsf{indr}}(\mathcal{A}) \leqslant \mathsf{Adv}_{\mathsf{F}}^{\mathsf{prf}}(\mathcal{A}_{\mathsf{prf}}) + \frac{1}{\sqrt{2}}\sqrt{\frac{S \cdot p \cdot q}{N}} \ .$$

Adversary $\mathcal{A}_{\mathsf{prf}}$ *is roughly as efficient as* $\mathcal{A}$.

Proof (of Theorem 4). Our proof begins with the PRF adversary $\mathcal{A}_{\mathsf{prf}}$ on the left side of Fig. 4. It simulates the view of $\mathcal{A}$ using its own oracle to provide $\mathcal{A}$ with the encryption of messages. Similarly the distinguisher $\mathcal{A}_{\mathsf{dist}}$ shown on the right side of Fig. 4 uses its sample oracle to simulate the view of $\mathcal{A}$.

The claim on the efficiency of $\mathcal{A}_{\mathsf{prf}}$ follow from examination of its code. Note that distinguisher $\mathcal{A}_{\mathsf{dist}}$ is S-bounded because it only needs to store the state of $\mathcal{A}$ during its oracle query (because M can be recomputed from this state).

We claim that the following equalities hold

$$\begin{aligned}
&\text{(i) } \Pr[\mathsf{G}_{\mathsf{F},1}^{\mathsf{prf}}(\mathcal{A}_{\mathsf{prf}})] = \Pr[\mathsf{G}_{\mathsf{CTR\$[F]},1}^{\mathsf{indr}}(\mathcal{A})],\\
&\text{(ii) } \Pr[\mathsf{G}_{\mathsf{F},0}^{\mathsf{prf}}(\mathcal{A}_{\mathsf{prf}})] = \Pr[\mathcal{A}_{\mathsf{dist}}(Y^q) \Rightarrow 1],\\
&\text{(iii) } \Pr[\mathcal{A}_{\mathsf{dist}}(X^q) \Rightarrow 1] = \Pr[\mathsf{G}_{\mathsf{CTR\$[F]},0}^{\mathsf{indr}}(\mathcal{A})].
\end{aligned}$$

Game $\mathsf{G}^{\mathsf{wprf}}_{\mathsf{F},b}(\mathcal{A})$	$\textsc{Ror}()$
$K \xleftarrow{\$} \mathsf{F.K}$	$X \xleftarrow{\$} \mathsf{F.Dom}$
$F \xleftarrow{\$} \mathsf{Fcs}(\mathsf{F.Dom}, \mathsf{F.Rng})$	$Y_1 \leftarrow \mathsf{F.Ev}(K, X)$
$b' \xleftarrow{\$} \mathcal{A}^{\textsc{Ror}}$	$Y_0 \leftarrow F(X)$
Return $b' = 1$	$Y_{-1} \xleftarrow{\$} \mathsf{F.Rng}$
	Return (X, Y_b)

Fig. 5. Games defining weak pseudorandom function security of a family of functions.

Here we let $X^q = \textsc{Rand}[N, 2^{\mathsf{F.ol}}, p, q]$ and $Y^q = \mathrm{CTR\$}[N, \mathsf{Fcs}(N, 2^{\mathsf{F.ol}}), p, q]$.

Claim (i) holds because in both games $\mathcal{A}$ is seeing encryptions of M using $\mathsf{CTR\$[F]}$. Claim (ii) holds because in both games $\mathcal{A}$ is seeing randomized counter-mode encryption of M using a random function F. Claim (iii) holds because in both games $\mathcal{A}$ is seeing random strings.

The calculations are then as follows.

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{CTR\$[F]}}(\mathcal{A}) &= \Pr[\mathsf{G}^{\mathsf{indr}}_{\mathsf{CTR\$[F]},1}(\mathcal{A})] - \Pr[\mathsf{G}^{\mathsf{indr}}_{\mathsf{CTR\$[F]},0}(\mathcal{A})] \\
&= \Pr[\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},1}(\mathcal{A}_{\mathsf{prf}})] - \Pr\left[\mathcal{A}_{\mathsf{dist}}(X^q) \Rightarrow 1\right] \\
&= \mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prf}}) - \mathsf{Adv}^{\mathsf{dist}}_{X^q, Y^q}(\mathcal{A}_{\mathsf{dist}}) \\
&\leqslant \mathsf{Adv}^{\mathsf{prf}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{prf}}) + \frac{1}{\sqrt{2}}\sqrt{\frac{S \cdot p \cdot q}{N}}\ .
\end{aligned}$$

The final inequality follows by applying Theorem 3 with the distinguisher that outputs the bit $1 \oplus \mathcal{A}^{\textsc{Samp}}_{\mathsf{dist}}$. □

Weak prf. Weak PRF security is a variant of PRF security where the game picks the input to the PRF at random for the adversary. Consider the game $\mathsf{G}^{\mathsf{wprf}}_{\mathsf{F},b}(\mathcal{A})$ shown in Fig. 5 when $b \in \{0,1\}$. The standard definition of WPRF security is $\mathsf{Adv}^{\mathsf{wprf}}_{\mathsf{F}}(\mathcal{A}) = \Pr[\mathsf{G}^{\mathsf{wprf}}_{\mathsf{F},1}(\mathcal{A})] - \Pr[\mathsf{G}^{\mathsf{wprf}}_{\mathsf{F},0}(\mathcal{A})]$. It asks that an adversary cannot distinguish between $\mathsf{F.Ev}(K, X)$ and $F(X)$ when X is picked at random and F is a random function.

For proofs a different version of WPRF security is preferable. Consider the game $\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},-1}(\mathcal{A})$. It differs from $\mathsf{G}^{\mathsf{wprf}}_{\mathsf{F},0}(\mathcal{A})$ because the $\textsc{Ror}$ oracle returns a fresh random Y even if X's repeat. We define the advantage of $\mathcal{A}$ by $\mathsf{Adv}^{\mathsf{wprf2}}_{\mathsf{F}}(\mathcal{A}) = \Pr[\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},1}(\mathcal{A})] - \Pr[\mathsf{G}^{\mathsf{prf}}_{\mathsf{F},-1}(\mathcal{A})]$. We call this WPRF2 security.

A family of functions is deterministic so its output will necessarily repeat on repeated inputs. Thus we can expect better security for the first definition. It is then useful to assume good WPRF security and have a generic proof that WPRF2 security cannot differ from it too much. It is straightforward to show, for example, that $|\mathsf{Adv}^{\mathsf{wprf}}_{\mathsf{F}}(\mathcal{A}) - \mathsf{Adv}^{\mathsf{wprf2}}_{\mathsf{F}}(\mathcal{A})| \leqslant q^2/N$. Using our space-bounded techniques we can show the following theorem which improves the bound when the space used by $\mathcal{A}$ is less than the number of queries it makes.

Lemma 10. *Let F be a family of functions with* $\mathsf{F.Dom} = [N]$. *Let $\mathcal{A}$ be an S-bounded adversary making at most q queries to its oracle. Then*

$$\left|\mathsf{Adv}^{\mathsf{wprf}}_{\mathsf{F}}(\mathcal{A}) - \mathsf{Adv}^{\mathsf{wprf2}}_{\mathsf{F}}(\mathcal{A})\right| \leqslant \frac{1}{\sqrt{2}}\sqrt{\frac{S \cdot q}{N}} \ .$$

Proof. First note that $|\mathsf{Adv}^{\mathsf{wprf}}_{\mathsf{F}}(\mathcal{A}) - \mathsf{Adv}^{\mathsf{wprf2}}_{\mathsf{F}}(\mathcal{A})| = |\mathsf{Pr}[\mathsf{G}^{\mathsf{wprf}}_{\mathsf{F},-1}] - \mathsf{Pr}[\mathsf{G}^{\mathsf{wprf}}_{\mathsf{F},0}(\mathcal{A})]|$ and suppose without loss of generality that this difference in probabilities is positive. Identify $\mathsf{F.Rng}$ with $[M]$. In game $\mathsf{G}^{\mathsf{wprf}}_{\mathsf{F},-1}$ the adversary is being given uniformly random samples $(X, Y) \xleftarrow{\$} [N] \times [M]$ and in game $\mathsf{G}^{\mathsf{wprf}}_{\mathsf{F},0}(\mathcal{A})$ it is seeing the same subject to the fact that Y will repeat whenever X does. These views are exactly identical to the view of a distinguisher in the setting of Theorem 3. Applying that result gives the state bound. ▭

4.4 CTR$ with a PRP and Weak PRPs

In practice most encryption uses AES - a blockcipher with domain $\{0, 1\}^{128}$ which is thus best modeled as a PRP. We do not know how to extend our $\mathsf{CTR\$}$ analysis for this case. Our streaming analysis with a random function F used that $H(F) = \log(M^N)$. If F is a random permutation then $H(F) = \log(N!)$ which is not sufficiently large. However, when only one block messages are encrypted, we can using the streaming problem addressed in Section 3 to bound the advantage by $O(\mathcal{O}_{\mathrm{sl}})$.

Security of $\mathsf{CTR\$}$ for one block messages corresponds closely to the WPRF2 security of the underlying blockcipher. Thus we divide the $\mathsf{CTR\$}$ proof into three steps. First we use Theorem 1 to obtain a bound in the streaming setting naturally induced by this problem. Next we use this to prove a generic "switching lemma" between Weak PRP (WPRP) security (defined momentarily) and WPRF2 security analogous to Lemma 10. The security of $\mathsf{CTR\$}$ for one block messages follows from this lemma in a straightforward way. The streaming analysis will be presented in full here. The WPRP and $\mathsf{CTR\$}$ results are stated, but the (straightforward) proofs are deferred to the full version of this paper [19].

Weak prp. WPRP security is defined via the games $\mathsf{G}^{\mathsf{wprp}}_{\mathsf{F},b}$ shown in Figure 6. The advantage of an adversary $\mathcal{A}$ against blockcipher F is defined by $\mathsf{Adv}^{\mathsf{wprp}}_{\mathsf{F}}(\mathcal{A}) = \mathsf{Pr}[\mathsf{G}^{\mathsf{wprp}}_{\mathsf{F},1}(\mathcal{A})] - \mathsf{Pr}[\mathsf{G}^{\mathsf{wprp}}_{\mathsf{F},0}(\mathcal{A})]$. The notion is essentially the same as for WPRF security, except the random function has been replaced with a random permutation.

The following lemma bounds the difference between an adversary's WPRP and WPRF2 advantages, allowing one to generically switch between the two. It is an almost immediate implication of the coming streaming analysis.

Lemma 11. *Let F be a family of functions with* $\mathsf{F.Dom} = \mathsf{F.Rng} = [N]$. *Let $\mathcal{A}$ be an S-bounded adversary making at most q queries to its oracle. Then*

$$\left|\mathsf{Adv}^{\mathsf{wprp}}_{\mathsf{F}}(\mathcal{A}) - \mathsf{Adv}^{\mathsf{wprf2}}_{\mathsf{F}}(\mathcal{A})\right| \leqslant 3\mathcal{O}_{\mathrm{sl}}(q, S, N) \ .$$

Game $\mathsf{G}^{\mathsf{wprp}}_{\mathsf{F},b}(\mathcal{A})$	$\textsc{Ror}()$
$K \xleftarrow{\$} \mathsf{F.K}$	$X \xleftarrow{\$} \mathsf{F.Dom}$
$F \xleftarrow{\$} \mathsf{Perm}(\mathsf{F.Dom})$	$Y_1 \leftarrow \mathsf{F.Ev}(K, X)$
$b' \xleftarrow{\$} \mathcal{A}^{\textsc{Ror}}$	$Y_0 \leftarrow F(X)$
Return $b' = 1$	Return (X, Y_b)

Fig. 6. Games for weak pseudorandom permutation security of a family of functions.

Randomized counter-mode. The following theorem (proved using Lemma 11) bounds the advantage of an attacker against $\mathsf{CTR\$}$ with a blockcipher by the WPRP security of the blockcipher when only one block messages are encrypted.

Theorem 5. *Let F be a blockcipher with $\mathsf{F.Dom} = \mathsf{F.Rng} = \{0,1\}^n$. Let $\mathcal{A}$ be an S-bounded adversary making at most q queries of length n to its oracle. Then we can build an adversary $\mathcal{A}_{\mathsf{wprp}}$ such that*

$$\mathsf{Adv}^{\mathsf{indr}}_{\mathsf{CTR\$[F]}}(\mathcal{A}) \leqslant \mathsf{Adv}^{\mathsf{wprp}}_{\mathsf{F}}(\mathcal{A}_{\mathsf{wprp}}) + 3\mathcal{O}_{\mathrm{sl}}(q, S, 2^n) \ .$$

Adversary $\mathcal{A}_{\mathsf{wprp}}$ is roughly as efficient as $\mathcal{A}$.

Steaming analysis. In the streaming setting we now analyze $\mathcal{A}$ is given repeated samples (R_i, P_i) where P_i is either random or $F(R_i)$ for a random $F \in \mathsf{Perm}(N)$. We first use $\mathcal{O}_{\mathrm{sl}}$ to switch to R_i being picked without replacement. Now $P_i = F(R_i)$ can be viewed as random samples without replacement; we use $\mathcal{O}_{\mathrm{sl}}$ again to switch P_i to being sampled with replacement. Then we use $\mathcal{O}_{\mathrm{sl}}$ a final time to switch R_i back to being picked with replacement.

Lemma 12. *Let N, q, and S be given. Further, let $W^q = \textsc{Rand}[N, N, 1, q]$ and $V^q = \mathrm{CTR\$}[N, \mathsf{Perm}(N), 1, q]$. Then for every S-bounded distinguisher $\mathcal{A}$, we have*

$$\mathsf{Adv}^{\mathsf{dist}}_{W^q, V^q}(\mathcal{A}) \leqslant 3\mathcal{O}_{\mathrm{sl}}(q, S, N) \ .$$

Proof. Consider the sequence of game G_0 through G_4 shown in Fig. 7.

In game G_0, each R_i is uniformly and independently sampled and $P_i = F(R_i)$ where F is a random permutation. This is exactly the distribution V^q so $\mathsf{Pr}[\mathsf{G}_0] = \mathsf{Pr}[\mathcal{A}(V^q) \Rightarrow 1]$. In game G_4, each R_i and each P_i are uniformly and independently sampled. This is exactly the distribution W^q so $\mathsf{Pr}[\mathsf{G}_4] = \mathsf{Pr}[\mathcal{A}(W^q) \Rightarrow 1]$. We can then see that,

$$\mathsf{Adv}^{\mathsf{dist}}_{W^q, V^q}(\mathcal{A}) = \sum_{i=1}^{4} \mathsf{Pr}[\mathsf{G}_i] - \mathsf{Pr}[\mathsf{G}_{i-1}]$$

Let X^q be sampling with replacement and Y^q be sampling without replacement from $[N]$. We will bound the difference between G_0 and G_4 by using a sequence of distinguishers for (X^q, Y^q), whose advantages we bound with $\mathcal{O}_{\mathrm{sl}}$.

Games G_0,G_1,G_2,G_3,G_4	$\underline{\textsc{Samp}()}$
$F \xleftarrow{\$} \mathsf{Perm}(N)$ // $\mathsf{G}_0, \mathsf{G}_1$	$R_i \xleftarrow{\$} [N]$ // G_0,G_4
$F \xleftarrow{\$} \mathsf{Fcs}(N, N)$ // G_2	$R_i \xleftarrow{\$} [N] \setminus \{R_1, \ldots, R_{i-1}\}$ // $\mathsf{G}_1, \mathsf{G}_2, \mathsf{G}_3$
$i \leftarrow 1$	$P_i \leftarrow F(R_i)$ // $\mathsf{G}_0, \mathsf{G}_1, \mathsf{G}_2$
$b' \xleftarrow{\$} \mathcal{A}^{\textsc{Samp}}$	$P_i \xleftarrow{\$} [N]$ // G_3,G_4
Return $b' = 1$	$i \leftarrow i + 1$
	Return (R_i, P_i)

Fig. 7. Games for proof of Lemma 12. Commented lines of code are only included in the indicated games.

The distinguishers are shown below, where $R_{<i} = \{R_1, \ldots, R_{i-1}\}$. As written, distinguishers $\mathcal{A}_{0,1}$ and $\mathcal{A}_{1,2}$ store large amounts of space. The former stores an entire random permutation $F : [N] \to [N]$. The latter stores a list of q different R_i values. Used naively, this would result in useless advantage bounds. However, note that the stored state is sampled *before any oracle queries are made*. Thus we can use a standard coin-fixing argument to upper bound the advantage of these distinguishers by the advantage of distinguishers $\mathcal{A}^*_{0,1}$ and $\mathcal{A}^*_{1,2}$ for which the best choices of F and the R_i values are hardcoded.

The description size of a distinguisher is not included in the bound of their state so we can see that $\mathcal{A}^*_{0,1}$ is S-bounded, $\mathcal{A}^*_{1,2}$ is S-bounded, and $\mathcal{A}_{3,4}$ is S-bounded. Note that $\mathcal{A}^*_{1,2}$ does not need to store the stage counter i for itself because this is provided as input as part of our streaming.

Distinguisher $\mathcal{A}_{0,1}^{\textsc{Samp}}$	Distinguisher $\mathcal{A}_{1,2}^{\textsc{Samp}}$	Distinguisher $\mathcal{A}_{3,4}^{\textsc{Samp}}$
$F \xleftarrow{\$} \mathsf{Perm}(N)$	For $i = 1, \ldots, q$ do	$b' \xleftarrow{\$} \mathcal{A}^{\textsc{SimEnc}}$
$b' \xleftarrow{\$} \mathcal{A}^{\textsc{SimSamp}}$	$\quad R_i \xleftarrow{\$} [N] \setminus R_{<i}$	Return b'
Return $1 \oplus b'$	$i \leftarrow 1$	
	$b' \xleftarrow{\$} \mathcal{A}^{\textsc{SimEnc}}$	$\underline{\textsc{SimSamp}()}$
$\underline{\textsc{SimSamp}()}$	Return b'	$R \leftarrow \textsc{Samp}$
$R \leftarrow \textsc{Samp}$		$P \xleftarrow{\$} [N]$
$P \leftarrow F(R)$	$\underline{\textsc{SimSamp}()}$	Return (R, P)
Return (R, P)	$P \leftarrow \textsc{Samp}$	
	$i \leftarrow i + 1$	
	Return (R_i, P)	

Now consider the transition from G_0 to G_1. They differ in whether R_i is sampled with or without replacement. Distinguisher $\mathcal{A}_{0,1}$ tries to use this difference to distinguish between X^q and Y^q using its samles to set R_i and simulating $P = F(R)$ for itself. We have $\Pr[\mathsf{G}_1] - \Pr[\mathsf{G}_0] = \mathsf{Adv}^{\mathsf{dist}}_{X^q, Y^q}(\mathcal{A}_{0,1})$. Note that $\mathcal{A}_{0,1}$ outputs the bit $1 \oplus b'$ to give the order we want.

Games G_1 and G_2 differ only in whether F is a random permutation or random function. Because they are being fed non-repeating input the values

$P_i = F(R_i)$ are distributed according to Y^q in the former case and X^q in the latter. Consequently, we can see that $\Pr[\mathsf{G}_2] - \Pr[\mathsf{G}_1] = \mathsf{Adv}^{\mathsf{dist}}_{X^q,Y^q}(\mathcal{A}_{1,2})$.

Games G_2 and G_3 are equivalent. They differ in whether each P_i is by $P_i \xleftarrow{\$} [N]$ or as $F(R_i)$ for a random function F. Because the R_i values are non-repeating these are the same distribution, giving $\Pr[\mathsf{G}_3] - \Pr[\mathsf{G}_2] = 0$.

Finally, G_3 and G_4 differ in whether R_i is sampled with or without replacement. Via $\mathcal{A}_{3,4}$ we again reduce this to distinguishing between X^q and Y^q. We have $\Pr[\mathsf{G}_4] - \Pr[\mathsf{G}_3] = \mathsf{Adv}^{\mathsf{dist}}_{X^q,Y^q}(\mathcal{A}_{3,4})$.

Plugging in to 4.4 and bounding with $\mathcal{A}^*_{0,1}$ and $\mathcal{A}^*_{1,2}$ gives

$$\mathsf{Adv}^{\mathsf{dist}}_{W^q,V^q}(\mathcal{A}) \leqslant \mathsf{Adv}^{\mathsf{dist}}_{X^q,Y^q}(\mathcal{A}^*_{0,1}) + \mathsf{Adv}^{\mathsf{dist}}_{X^q,Y^q}(\mathcal{A}^*_{1,2}) + \mathsf{Adv}^{\mathsf{dist}}_{X^q,Y^q}(\mathcal{A}_{3,4}) \ .$$

The result follows by bounding these advantages with $\mathcal{O}_{\mathrm{sl}}$. □

4.5 Other results

Encrypt-then-prf. In the full version of this paper [19] we apply the above result to the proving the security of the encrypt-then-PRF construction of an authenticated encryption scheme (for fixed length messages).

Nonce-based encryption. We note that our $\mathsf{CTR\$}$ and encrypt-then-prf theorems composes correctly with the standard way of constructing nonce-based encryption from a randomized encryption scheme by applying a PRF to the nonce to obtain coins for the underlying encryption scheme.

Other encryption schemes. In the full version of this paper [19] we look at streaming models induced by other randomized encryption schemes (CTR$ with a permutation, OFB$, CBC$, and CFB$). We exhibit straightforward attacks which distinguish length $p \in \Theta(\sqrt{N})$ samples from random with low state, $q = 1$, and good advantage.

Our streaming proof for the model induced by CTR$ with a random function implies such an attack is not possible against it. However, to be clear, these attacks *do not* rule out good time-memory tradeoffs for these other schemes. Instead these very weak attacks indicate that if such bounds are possible, their proofs will require new insights/models. See the full version of this paper [19] for more discussion.

5 Open Questions

Our work leaves open a number of important questions - most directly resolving validity of Conjecture 1 (or a relaxed version thereof which suffices for our final statement). More generally, there is the question of which other encryption schemes admit proofs of tight time-memory trade-offs. Furthermore, we do not know how to prove trade-offs for more complex security games which do not fit within the streaming model, e.g., security in the presence of decryption oracles.

Acknowledgements

We thank Aishwarya Thiruvengadam for insightful discussions in the initial stage of this project. Jaeger was supported in part by NSF grants CNS-1717640 and CNS-1526801, and by NSF grant CNS-1553758 while visiting UC Santa Barbara.

Stefano Tessaro's work was partially supported by NSF grants CNS-1553758 (CAREER), CNS-1719146, CNS-1528178, and IIS-1528041, and by a Sloan Research Fellowship.

References

1. B. M. Abrego, S. Fernandez-Merchant, M. G. Neubauer, and W. Watkins. Sum of squares of degrees in a graph. *Journal of Inequalities in Pure and Applied Mathematics*, 10(3), 2009.
2. Joël Alwen, Binyi Chen, Krzysztof Pietrzak, Leonid Reyzin, and Stefano Tessaro. Scrypt is maximally memory-hard. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part III*, volume 10212 of *LNCS*, pages 33–62. Springer, Heidelberg, April / May 2017.
3. Joël Alwen and Vladimir Serbinenko. High parallel complexity graphs and memory-hard functions. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *47th ACM STOC*, pages 595–603. ACM Press, June 2015.
4. 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.
5. SH Babbage. Improved "exhaustive search" attacks on stream ciphers. *European Convention on Security and Detection*, pages 161–166, May 1995.
6. Elad Barkan, Eli Biham, and Adi Shamir. Rigorous bounds on cryptanalytic time/memory tradeoffs. In Cynthia Dwork, editor, *CRYPTO 2006*, volume 4117 of *LNCS*, pages 1–21. Springer, Heidelberg, August 2006.
7. Mihir Bellare, Ted Krovetz, and Phillip Rogaway. Luby-Rackoff backwards: Increasing security by making block ciphers non-invertible. In Kaisa Nyberg, editor, *EUROCRYPT'98*, volume 1403 of *LNCS*, pages 266–280. Springer, Heidelberg, May / June 1998.
8. 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.
9. Christian Bey. An upper bound on the sum of squares of degrees in a hypergraph. *Discrete Math.*, 269(1-3):259–263, July 2003.
10. Sebastian M. Cioab. Note: Sums of powers of the degrees of a graph. *Discrete Math.*, 306(16):1959–1964, August 2006.
11. Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, New York, NY, USA, 2006.
12. 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.

13. Donald W. Davies and Graeme I. P. Parkin. The average cycle size of the key-stream in output feedback encipherment. In Thomas Beth, editor, *EUROCRYPT'82*, volume 149 of *LNCS*, pages 263–279. Springer, Heidelberg, March / April 1983.
14. D. de Caen. An upper bound on the sum of squares of degrees in a graph. *Discrete Math.*, 185(1-3):245–248, April 1998.
15. Jovan Dj. Golic. Cryptanalysis of alleged A5 stream cipher. In Walter Fumy, editor, *EUROCRYPT'97*, volume 1233 of *LNCS*, pages 239–255. Springer, Heidelberg, May 1997.
16. Vytautas Gruslys, Shoham Letzter, and Natasha Morrison. Hypergraph lagrangians: Resolving the frankl-füredi conjecture. *arXiv preprint arXiv:1807.00793*, 2018.
17. Tetsu Iwata. New blockcipher modes of operation with beyond the birthday bound security. In Matthew J. B. Robshaw, editor, *FSE 2006*, volume 4047 of *LNCS*, pages 310–327. Springer, Heidelberg, March 2006.
18. Tetsu Iwata, Bart Mennink, and Damian Vizár. CENC is optimally secure. Cryptology ePrint Archive, Report 2016/1087, 2016. http://eprint.iacr.org/2016/1087.
19. Joseph Jaeger and Stefano Tessaro. Tight time-memory trade-offs for symmetric encryption. Cryptology ePrint Archive, Report 2019/???, 2019. https://eprint.iacr.org/2019/???
20. Vladimir Nikiforov. Note: The sum of the squares of degrees: Sharp asymptotics. *Discrete Math.*, 307(24):3187–3193, November 2007.
21. Noam Nisan. Pseudorandom generators for space-bounded computation. *Combinatorica*, 12(4):449–461, 1992.
22. Jacques Patarin. Mirror theory and cryptography. Cryptology ePrint Archive, Report 2016/702, 2016. http://eprint.iacr.org/2016/702.
23. J. M. Pollard. A monte carlo method for factorization. *BIT Numerical Mathematics*, 15(3):331–334, Sep 1975.
24. Jean-Jacques Quisquater and Jean-Paul Delescaille. How easy is collision search. New results and applications to DES. In Gilles Brassard, editor, *CRYPTO'89*, volume 435 of *LNCS*, pages 408–413. Springer, Heidelberg, August 1990.
25. 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.
26. 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.