On the Streaming Indistinguishability of a Random Permutation and a Random Function

Itai Dinur

Department of Computer Science, Ben-Gurion University, Israel

Abstract. An adversary with S bits of memory obtains a stream of Q elements that are uniformly drawn from the set $\{1, 2, \ldots, N\}$, either with or without replacement. This corresponds to sampling Q elements using either a random function or a random permutation. The adversary's goal is to distinguish between these two cases.

This problem was first considered by Jaeger and Tessaro (EUROCRYPT 2019), which proved that the adversary's advantage is upper bounded by $\sqrt{Q \cdot S/N}$. Jaeger and Tessaro used this bound as a streaming switching lemma which allowed proving that known time-memory tradeoff attacks on several modes of operation (such as counter-mode) are optimal up to a factor of $O(\log N)$ if $Q \cdot S \approx N$. However, the bound's proof assumed an unproven combinatorial conjecture. Moreover, if $Q \cdot S \ll N$ there is a gap between the upper bound of $\sqrt{Q \cdot S/N}$ and the $Q \cdot S/N$ advantage obtained by known attacks.

In this paper, we prove a tight upper bound (up to poly-logarithmic factors) of $O(\log Q \cdot Q \cdot S/N)$ on the adversary's advantage in the streaming distinguishing problem. The proof does not require a conjecture and is based on a hybrid argument that gives rise to a reduction from the unique-disjointness communication complexity problem to streaming.

Keywords: Streaming algorithm, time-memory tradeoff, communication complexity, provable security, switching lemma, mode of operation.

1 Introduction

A classical result in cryptography asserts that an adversary attempting to distinguish a random permutation from a random function with an image size of N using Q queries has advantage that is upper bounded by about Q^2/N over a coin toss [3, 14, 15]. This bound serves as a switching lemma which has important implications in establishing the security of various cryptographic constructions. For example, the security of several modes of operation (such as counter-mode) is proved up to the birthday bound of $Q = \sqrt{N}$ by first idealizing the underlying block cipher as a random permutation and then replacing it with a random function using the switching lemma.[1]

[1] For the sake of brevity, in this paper we use the term "switching lemma" to refer to a particular type of lemma that allows to switch between a random permutation and a random function.

A limitation of the switching lemma is that it only bounds the advantage of the adversary as a function of the number of queries, whereas in practice, the adversary could have constraints on additional resources, notably on memory. At the same time, given $Q \approx \sqrt{N}$ unrestricted queries to the underlying primitive, it is possible to distinguish a random function from a random permutation with constant advantage using a negligible amount of $O(\log N)$ bits of memory by applying a "memory-less" cycle detection algorithm such as Floyd's algorithm [18] (or its variants, e.g., [7, 22]).

Streaming Indistinguishability Cycle detection algorithms are inapplicable when only given access to a stream of data produced by arbitrary queries to the underlying primitive which are not under the adversary's control. The streaming indistinguishability model was introduced in the context of symmetric-key cryptography by Jaeger and Tessaro at EUROCRYPT 2019 [16]. The authors considered an adversary (i.e. a randomized algorithm) with memory size of S bits and access to a stream of Q elements drawn from either a random permutation or from a random function with an image size of N. The main technical result of [16] is an adaptation of the switching lemma between a random permutation and random function to the streaming model. The streaming switching lemma asserts that the adversary's advantage is bounded by $\sqrt{Q \cdot S/N}$ as long as the queries to the underlying primitive are not repeated. The proof of the bound is based on tools from information theory and relies on a combinatorial conjecture regarding hypergraphs. We refer the reader to [16] for more details.

The main applications of the switching lemma described in [16] deal with cryptanalysis of modes of operations. Such modes are typically secure up to the birthday bound against adversaries with unbounded memory, yet [16] shows that they become more secure against memory-bounded adversaries. For example, in AES-based randomized counter-mode, message m_i is encrypted as $r_i, c_i = \mathrm{AES}_K(r_i) \oplus m_i$, where r_i is a random 128-bit string. The best known distinguishing attack simply awaits a collision $r_i = r_j$ for $i \neq j$, in which case $c_i \oplus c_j = m_i \oplus m_j$. This attack stores the r_i's and requires memory of about $\sqrt{N} = 2^{64}$ to find a collision with high probability. Let us now assume that the memory is limited to storing only $S' \ll 2^{64}$ values (where $S' \approx S \cdot \log N$ bits, as storing an element requires $\log N$ bits). In this case, the probability of observing a collision with a stored element (i.e., the distinguishing advantage) is roughly $Q \cdot S'/N \approx Q \cdot S/N$ (ignoring a logarithmic factor in N). Hence, such a collision is likely to occur only after observing about $Q \approx N/S \gg 2^{64}$ elements.

Jaeger and Tessaro used their streaming switching lemma to show that the simple attack on randomized counter-mode describe above is optimal up to a factor of $O(\log N)$, if we require a constant advantage. The proof applies the streaming switching lemma to replace the random r_i's with random non-repeating ones and further replaces AES with a truly random permutation (assuming it is a PRP). Finally, it applies the streaming switching lemma again to replace the permutation with a random function, completely masking the messages. More details and additional applications are described in [16]. We further mention that attacks against counter-mode and other modes of operation have been shown to be meaningful in practice (refer to [4] for a recent example), giving an additional motivation to understand their limitations.

The streaming switching lemma of [16] is very useful, but has two limitations. First, it is based on an unproven combinatorial conjecture. Second, when $Q \cdot S \ll N$, there is a gap between the advantage upper bound $\sqrt{Q \cdot S/N}$ of the lemma and the $Q \cdot S/N$ advantage of the simple attack described above. In fact, it is easy to see that the bound $\sqrt{Q \cdot S/N}$ is not tight when $Q \cdot S \ll N$ and $S \approx Q$, as it evaluates to $Q/\sqrt{N}$. On the other hand, the true optimal advantage is Q^2/N, as obtained by the original switching lemma (since for $S \approx Q$, the adversary can store all the elements in the stream).

In order to demonstrate this gap, let us assume that for $N = 2^{128}$ the adversary has memory limited to storing $S = 2^{40}$ elements, and obtains a stream of $Q = 2^{64}$ elements. Jaeger and Tessaro's result upper bounds the adversary's advantage by about $\sqrt{2^{64+40-128}} = 2^{-12}$. On the other hand, the distinguishing advantage of the attack described above is $2^{64+40-128} = 2^{-24}$, which is significantly lower.

Our Results In this paper, we overcome the two limitations of Jaeger and Tessaro's result. More specifically, we derive a streaming switching lemma which bounds the adversary's advantage by $O(\log Q \cdot Q \cdot S/N)$ via an alternative proof which it is not based on any conjecture. This matches the advantage of the simple distinguishing attack described above (up to poly-logarithmic factors in N), hence we resolve the streaming indistinguishability problem unconditionally.[2] Note that if we plug $S = Q$ into our bound, we obtain the original switching lemma (up to poly-logarithmic factors). Hence, our bound can also be viewed as a natural generalization of the original switching lemma to the case that the adversary cannot store all the Q elements of the stream (i.e. $S \ll Q$).

Finally, we extend the streaming switching lemma to show that the advantage of an adversary with S bits of memory that is allowed P passes over a stream of Q elements (drawn from a random permutation or a random function) is bounded by $O(\log Q \cdot Q \cdot S \cdot P/N)$. If we combine the multi-pass bound with the original switching lemma, we obtain the bound of about $\min\{\log Q \cdot Q \cdot S \cdot P/N, Q^2/N\}$, which is tight up to poly-logarithmic factors in N.

To understand the significance of our multi-pass bound, observe that for a fixed value of S, the P-pass streaming bound depends only on the total number of queries, $Q \cdot P$ (ignoring the small factor of $\log Q$). This essentially implies that repeating Q distinct queries P times does not give a P-pass algorithm an advantage over a single-pass algorithm that issues $Q \cdot P$ distinct queries. In contrast, in the non-streaming model repeating queries in an adaptive way has a big advantage, as cycle detection algorithms perform significantly better than the P-pass bound (obtaining constant advantage for $S = O(\log N)$ and $\sqrt{N}$ queries).

[2] We note, however, that Jaeger and Tessaro's result is superior to ours by a factor of up to $O(\sqrt{\log Q})$ when $S \cdot Q \approx N$.

Our Techniques The main novelty of the proof of our switching lemma is a hybrid argument that allows to devise a reduction from communication complexity to streaming. The hybrid argument is tailored to a common cryptographic setting where the goal is to distinguish between two pre-fixed distributions on streams. The cryptographic setting is different from the typical worst-case setting of streaming problems, where there is much more freedom in choosing the stream distributions in reductions from communication complexity, and hybrid arguments are not required. Although it is simple, this hybrid argument is somewhat non-trivial and allows us to apply strong bounds from communication complexity to the problem. This proof naturally extends to multi-pass adversaries. On the other hand, it seems challenging to extend the proof of [16] to multi-pass adversaries, where queries to the underlying primitive are repeated. This further demonstrates that our proof technique may be of independent interest.

Related Work This work lies in the intersection between cryptography and streaming algorithms. The area of streaming algorithms is subject to active research in computer science, and has been largely influenced by the seminal work of Alon, Matias, and Szegedy on approximating frequency moments with limited space [1]. In the field of cryptography, several previous works investigated the security of cryptographic primitives against a space-bounded adversary whose input is given as a data stream composed of a sequence of elements that can be read only once (cf., [8, 21]). More recently, Thiruvengadam and Tessaro initiated the study of the security of modes of operation against space-bounded adversaries [24]. Jaeger and Tessaro's work [16], as well as this paper, continue the line of research on streaming algorithms in cryptography.

Paper Organization The rest of the paper is organized as follows. We give a technical overview of the proof in Section 2 and describe preliminaries in Section 3. In Section 4 we prove our main streaming switching lemma for single-pass algorithms, while our proof of the multi-pass variant is given in Section 5. Finally, we conclude the paper in Section 6.

2 Technical Overview

We consider an algorithm with S bits of memory that processes a stream of $Q \leq N$ elements from $[N] = \{1, 2, \ldots, N\}$, element by element. The goal of the algorithm is to decide whether the stream is drawn from a random permutation (i.e., the elements are drawn uniformly without replacement), or from a random function (i.e., the elements are drawn uniformly with replacement).

In [16] Jaeger and Tessaro approached the problem by considering the sequences of states maintained by the adversary for the two stream distributions, claiming that they remain statistically close.

In the rest of this section, we give an overview of our proof, which (unlike Jaeger and Tessaro's proof) does not directly analyze the states maintained by the adversary. For the sake of simplicity, in this overview we aim to show that the distinguishing advantage of any algorithm (compared to a random guess) is negligible as long as $Q \ll N/S$, but do not consider the concrete advantage.

2.1 Communication Complexity

A standard approach for obtaining bounds on streaming algorithms is via a reduction from communication complexity. Suppose that our goal is to distinguish between two distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ on a stream $x_1, x_2, \ldots, x_Q \in [N]^Q$. We can reduce the problem from a 2-player communication game between $\mathcal{A}$ and $\mathcal{B}$ as follows. For some value of i, we partition the stream into two parts, $x_1, \ldots, x_i$ and $x_{i+1}, \ldots, x_Q$. We give the first part to $\mathcal{A}$ and the second part to $\mathcal{B}$. The goal of $\mathcal{A}$ and $\mathcal{B}$ is to decide whether the (concatenated) stream is drawn from $\mathcal{D}_1$ or from $\mathcal{D}_2$ with minimal one-way communication between $\mathcal{A}$ and $\mathcal{B}$.

In the reduction, $\mathcal{A}$ simulates a streaming algorithm on its input, sends its intermediate state to $\mathcal{B}$, which continues the simulation of the streaming algorithm and outputs its result. Thus, any streaming algorithm with memory S yields a one-way communication protocol with communication cost of S and the same distinguishing advantage. Therefore, an upper bound on the distinguishing advantage of $\mathcal{A}$ and $\mathcal{B}$ in any one-way communication protocol yields a bound on the distinguishing advantage of any streaming algorithm.

Obviously, in order to obtain a meaningful upper bound on the distinguishing advantage in the communication game, the communication problem induced from the streaming problem must be hard. In particular, a reduction from communication complexity to the streaming distinguishability game could be useful only if it has the property that for both stream distributions considered in the game, each player receives an input (partial stream) drawn from the same marginal distribution. Otherwise, a player could trivially distinguish between the two distributions locally with no communication (since $\mathcal{A}$ and $\mathcal{B}$ are unrestricted computationally).

Suppose that $\mathcal{D}_1$ is the distribution where $x_1, x_2, \ldots, x_Q$ are sampled using a random permutation, and $\mathcal{D}_2$ is the distribution where the elements are sampled using a random function. Unfortunately, for $Q > 2$ there is no way to partition the stream between $\mathcal{A}$ and $\mathcal{B}$ such that each player receives an input with the same marginal distribution in both cases.

In order to work around this difficulty, we define hybrid stream distributions between $\mathcal{D}_1$ and $\mathcal{D}_2$ with the aim of bounding the advantage between each pair of neighboring distributions using communication complexity, and applying a hybrid argument to bound the total advantage.

2.2 An Initial Approach

We start by informally outlining an initial approach that does not give the desired bound, but motivates the alternative approach that follows. We denote a stream drawn from a random permutation by $x_1, \ldots, x_Q$ and a stream drawn from a random function by $\hat{x}_1, \ldots, \hat{x}_Q$. We define $Q - 1$ intermediate stream distributions, which give rise to Q distinguishing games. The i'th game involves distinguishing between the stream distributions

$$x_1, \dots, x_{Q-i}, \hat{x}_{Q-i+1}, \dots, \hat{x}_Q \text{ and } x_1, \dots, x_{Q-i-1}, \hat{x}_{Q-i}, \dots, \hat{x}_Q,$$

which is equivalent to distinguishing between

$$x_1, \dots, x_{Q-i} \text{ and } x_1, \dots, x_{Q-i-1}, \hat{x}_{Q-i}.$$

Namely, the goal is to determine whether the last element already appears in the stream or not. In fact, even if the last element is chosen uniformly, it will not appear in the stream with probability $1-(Q-i-1)/N$. Hence, we can condition on the event that $\hat{x}_{Q-i}$ appears in the stream. As a result, the distinguishing advantage of any algorithm can be approximately bounded by $\alpha \cdot (Q-i-1)/N$, where $\alpha = \alpha(i)$ is the advantage of the algorithm in distinguishing between $x_1, \dots, x_{Q-i}$ and $x_1, \dots, x_{Q-i-1}, \hat{x}_{Q-i}$, where $\hat{x}_{Q-i}$ is drawn uniformly from the first $Q-i-1$ elements of the stream.

Unfortunately, this approach is insufficient to prove the bound we require via a hybrid argument (regardless of whether we use communication complexity of any other tool). In order to demonstrate this, consider the following distinguishing algorithm that uses only $O(\log N)$ bits of memory: we iteratively hash every element of $x_1, \dots, x_{Q-i-1}$ to a single bit, maintaining the majority of the hashes. Then, we hash the final element and output 1 if and only if its hash is equal to the majority over the first $Q-i-1$ hashes. Simple calculation shows that the advantage of the algorithm in distinguishing between the above streams is about $\alpha = 1/\sqrt{Q-i-1}$. This implies that using this method cannot give a better upper bound than $1/\sqrt{Q-i-1} \cdot (Q-i-1)/N$ on the advantage of a streaming algorithm with memory $S = O(\log N)$ in distinguishing between neighboring stream distributions. If we sum over the advantages of the first $Q-1$ games (the advantage is 0 in the last game), we obtain

$$\sum_{i=0}^{Q-2} \frac{1}{\sqrt{Q-i-1}} \cdot \frac{Q-i-1}{N} = \sum_{i=0}^{Q-2} \frac{\sqrt{Q-i-1}}{N} = \Omega\left(\frac{Q^{3/2}}{N}\right),$$

which is already $\Omega(1)$ for $Q = N^{2/3}$. On the other hand, our goal is to show that if $S = O(\log N)$ and the distinguishing advantage is $\Omega(1)$, then $Q \approx N$.

2.3 The Improved Approach

The reason that the initial attempt above fails to prove the required bound is that distinguishing neighboring stream distributions is too easy, and the sum of the advantages over all Q games results in a loose bound. An alternative approach in attempt to overcome the loss is to try and avoid the straightforward sum of advantages by using more advanced techniques developed in the area of provable security for the purpose of obtaining tight bounds (e.g., the chi-squared method proposed in [11]). However, such techniques do not directly apply to the streaming model where the adversary no longer has access to answers of its previous queries. Moreover, it seems challenging to extend such techniques to the multi-pass setting in order to handle the dependencies between repeated queries to the underlying primitive. In this paper, we use a completely different approach by reconsidering our definition of intermediate hybrid distributions that lead from a stream produced by random permutation to a stream produced by a random function.

The Hybrid Distributions We start by defining the first distinguishing game between $x_1, \ldots, x_Q$ (a stream drawn from a random permutation) and a second stream drawn from a carefully chosen hybrid distribution. Our goal is to make sure that the distinguishing advantage between two neighboring stream distributions is significantly lower compared to the basic approach. Furthermore, we would like to use communication complexity in order to analyze neighboring stream distributions, i.e., we require that the stream can be partitioned such that the marginal distributions of the inputs given to each player are identical.

We define our stream distributions using more convenient notation of $x_1, \ldots, x_{Q/2}, y_1, \ldots, y_{Q/2}$, where each of $x_1, \ldots, x_{Q/2}$ and $y_1, \ldots, y_{Q/2}$ is a stream drawn from a random permutation, such that the streams are either drawn from the same permutation (which corresponds to the original distribution), or drawn from independent permutations (which corresponds to the first intermediate hybrid). We then define the corresponding 2-player communication problem (which we call the *permutation-dependence* problem), where $\mathcal{A}$ and $\mathcal{B}$ obtain $x_1, \ldots, x_{Q/2}$ and $y_1, \ldots, y_{Q/2}$, respectively, and try to decide with minimal one-way communication whether their inputs are drawn from the same or from independent permutations.

To complete the distinguishability upper bound proof for the streaming game, we prove an upper bound on the distinguishing advantage of $\mathcal{A}$ and $\mathcal{B}$ in the permutation-dependence problem. The proof is by a reduction from the *set-disjointness* problem, which is a canonical 2-player problem in communication complexity [2, 17, 23], where the input of each player is a set and their goal is to determine whether their sets intersect, or are disjoint.[3]

The first hybrid breaks the dependency between the two halves of the stream. We can now continue recursively by dividing the halves into quarters, etc. This results in a binary tree of hybrids of hight $\log Q$, where a one-way communication game is played at every non-leaf node. The leaves are completely independent elements of $[N]$, whose concatenation is a stream sampled using a random function, as desired.[4]

[3] In fact, the reduction is from the *unique-disjointness* problem which is a variant of set-disjointness with the promise that if the sets of the players intersect, the intersection size is 1.

[4] A hybrid argument on a binary tree is also used to prove the security of the classical pseudo-random function construction by Goldreich et al. [12]. However, the resemblance is superficial, as in [12] the construction itself is a binary tree, whereas in our case, we build it artificially only in the proof.

Summing up the advantages over the hybrids in each level of the tree gives an upper bound of $O(Q \cdot S/N)$. The overall advantage is $O(\log Q \cdot Q \cdot S/N)$, as there are $\log Q$ levels in the tree.

3 Preliminaries

Unless stated explicitly, all parameters considered in this paper are positive integers. We define $[N] = \{1, 2, \ldots, N\}$ and $[N]^K = \underbrace{[N] \times [N] \times \ldots \times [N]}_{K}$. Given bit strings x and y, we denote their concatenation by $x\|y$. For a positive integer K, we denote by $x^{(K)}$ the string $\underbrace{x\|x \ldots \|x}_{K}$, obtained by K repetitions of x. We denote by $HW(x)$ the Hamming weight of x.

Given a bit string $a \in \{0,1\}^N$ such that $HW(a) = K$, we can treat it as an incidence vector of a set $\{x_1, x_2, \ldots, x_K\}$ such that $x_i \in [N]$ and $a[x_i] = 1$ for $i \in [K]$. We define $SEQ : \{0,1\}^N \to [N]^K$ as the sequence $SEQ(a) = x_1, x_2, \ldots, x_K$ (which includes the elements indicated by a in lexicographical order). Given incidence vectors $a \in \{0,1\}^N$ and $b \in \{0,1\}^N$, let $a \cap b$ denote the intersection of these sets, and $|a \cap b|$ the size of the intersection.

Given a distribution $\mathcal{X}$ on strings with finite support, we write $x \overset{\$}{\leftarrow} \mathcal{X}$ to denote a random variable x chosen from $\mathcal{X}$. We write $x \sim \mathcal{X}$ if x is a random variable that is distributed as $\mathcal{X}$.

For arbitrary distributions on strings $\mathcal{D}_1$ and $\mathcal{D}_2$, we denote by $\mathcal{D}_1\|\mathcal{D}_2$ the distribution on strings obtained by concatenating two strings sampled independently from $\mathcal{D}_1$ and $\mathcal{D}_2$.

Distinguishing between Streams We define our model for a randomized algorithm whose goal is to distinguish between streams. The model is similar to the one defined in [16], although we use slightly different notation.

For some parameters N, K, let $\mathcal{X}$ be some distribution over $[N]^K$. We denote by $\mathrm{O}(\mathcal{X})$ an oracle that samples $x_1, x_2, \ldots, x_K$ from $\mathcal{X}$. The oracle receives up to K queries and answers query number i by x_i. Note that once the oracle outputs x_i, it is not output again. This implies that an algorithm $\mathcal{A}$ that interacts with $\mathrm{O}(\mathcal{X})$ receives $x_1, x_2, \ldots, x_K$ as a stream, i.e., if $\mathcal{A}$ requires access to x_i after issuing query i, it has to store x_i in memory in some representation.

We denote by $\mathcal{A}^{\mathrm{O}(\mathcal{X})}$ a randomized algorithm with oracle access to $\mathrm{O}(\mathcal{X})$ and by $\mathcal{A}^{\mathrm{O}(\mathcal{X})} \Rightarrow b$ the event that the algorithm outputs the bit $b \in \{0,1\}$.

We say that an algorithm $\mathcal{A}$ is S-bounded, if the size of each state maintained by $\mathcal{A}$ during any execution is upper bounded by S bits.

Let $\mathcal{X}$ and $\mathcal{Y}$ be two distributions over $[N]^K$. The streaming distinguishing advantage of an algorithm $\mathcal{A}$ between $\mathcal{X}$ and $\mathcal{Y}$ is defined as

$$\mathrm{Adv}^{\mathrm{STR}}_{\mathcal{X},\mathcal{Y}}(\mathcal{A}) = \left|\Pr[\mathcal{A}^{\mathrm{O}(\mathcal{X})} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathrm{O}(\mathcal{Y})} \Rightarrow 1]\right|.$$

We further define the optimal advantage for an S-bounded algorithm as

$$\mathrm{Opt}^{\mathrm{STR}}_{\mathcal{X},\mathcal{Y}}(S) = \max_{\mathcal{A}}\{\mathrm{Adv}^{\mathrm{STR}}_{\mathcal{X},\mathcal{Y}}(\mathcal{A}) \mid \mathcal{A} \text{ is } S-\text{bounded}\}.$$

Sampling with and without Replacement For a parameter $0 < K \leq N$, let $\mathcal{D}_N^K$ be the distribution over $[N]^K$ that is defined by a sampling procedure which uniformly draws K elements from $[N]$ without replacement.

For parameters $0 < K \leq N$ and $R > 0$, let $\mathcal{D}_N^{K \times R}$ be the distribution over $[N]^{K \cdot R}$ that is composed of R independent copies of $\mathcal{D}_N^K$. For example, $\mathcal{D}_N^{K \times 2} = \mathcal{D}_N^K \| \mathcal{D}_N^K$.

Note that sampling from $\mathcal{D}_N^{1 \times K}$ is equivalent to choosing K items from $[N]$ uniformly with replacement (i.e., from a random function), while sampling from $\mathcal{D}_N^K$ is equivalent to choosing K items from $[N]$ uniformly without replacement (i.e., from a random permutation).

The original switching lemma between a random permutation and a random function [3, 14, 15] asserts that any algorithm that issues Q queries to the underlying primitive has distinguishing advantage bounded by $Q^2/2N$. This bound obviously holds in the (more restricted) streaming model.

Theorem 1 (switching lemma [3, 14, 15]). *For any S and $Q \leq N$,*

$$\mathrm{Opt}^{\mathrm{STR}}_{\mathcal{D}_N^Q, \mathcal{D}_N^{1 \times Q}}(S) \leq \frac{Q^2}{2N}.$$

The Set-Disjointness and Unique-Disjointness Problems

The set-disjointness function $DISJ : \{0,1\}^N \times \{0,1\}^N \to \{0,1\}$ is defined as

$$DISJ(a,b) = \begin{cases} 0, & \text{there exists } i \in [N] \text{ for which } a[i] = b[i] = 1 \\ 1, & \text{otherwise.} \end{cases}$$

We can view a and b as subsets of $[N]$, encoded as incidence vectors, and then $DISJ(a,b) = 1$ if a and b are disjoint.

The *set-disjointness problem* (or *disjointness* in short) is a classical problem in communication complexity.[5] We consider its 2-player variant which is a game between $\mathcal{A}$ and $\mathcal{B}$ that run a protocol Π. In an instance of disjointness $\mathcal{A}$ receives $a \in \{0,1\}^N$, $\mathcal{B}$ receives $b \in \{0,1\}^N$ and their goal is to output $DISJ(a,b)$ with minimal communication in the worst case. Namely, the communication cost of Π is defined as the maximal number of bits communicated among all possible protocol executions.

We consider a variant of the disjointness problem called *unique-disjointness*, which is identical to disjointness, but with the promise that in a 0-instance, there exists a *single* index $i \in [N]$ for which $a[i] = b[i] = 1$. We denote the corresponding function by $UDISJ$, where we define $UDISJ(a,b) = \perp$ if a, b do not satisfy

[5] For a (slightly outdated) survey on set-disjointness, refer to [9].

the required promise. We will be interested in a public-coin randomized variant of unique-disjointness in which $\mathcal{A}, \mathcal{B}$ have access to a shared random string that is independent of their inputs.

We denote the output of the protocol Π on inputs a, b as $UDISJ_{\Pi}(a, b)$. Note that it is a random variable that depends on the shared randomness of $\mathcal{A}, \mathcal{B}$. Disjointness and its variants are worst case problems. This motivates the following notation for the error and advantage of the protocol.[6]

$$\begin{aligned}
\mathrm{Err}_N^{\mathrm{UDISJ0}}(\Pi) &= \max_{a,b}\{\Pr[UDISJ_{\Pi}(a,b) \neq 0 \mid UDISJ(a,b) = 0]\}, \\
\mathrm{Err}_N^{\mathrm{UDISJ1}}(\Pi) &= \max_{a,b}\{\Pr[UDISJ_{\Pi}(a,b) \neq 1 \mid UDISJ(a,b) = 1]\}, \\
\mathrm{Err}_N^{\mathrm{UDISJ}}(\Pi) &= \max\{\mathrm{Err}_N^{\mathrm{UDISJ0}}(\Pi), \mathrm{Err}_N^{\mathrm{UDISJ1}}(\Pi)\}, \\
\mathrm{Adv}_N^{\mathrm{UDISJ}}(\Pi) &= \left|1 - \mathrm{Err}_N^{\mathrm{UDISJ1}}(\Pi) - \mathrm{Err}_N^{\mathrm{UDISJ0}}(\Pi)\right|.
\end{aligned}$$

The following is a classical result in communication complexity.

Theorem 2 ([2, 17, 23, adapted]). *Any public-coin randomized protocol Π that solves unique-disjointness on all inputs $a, b \in \{0,1\}^N \times \{0,1\}^N$ such that $UDISJ(a, b) \in \{0, 1\}$ with error probability $\mathrm{Err}_N^{\mathrm{UDISJ}}(\Pi) \leq 1/3$, uses $\Omega(N)$ bits of communication in the worst case.*

Therefore, it is not possible to do much better than the trivial protocol in which $\mathcal{A}$ sends $\mathcal{B}$ its entire input a, and $\mathcal{B}$ outputs $UDISJ(a, b)$.

When analyzing the advantage γ of a protocol with communication cost of $o(N)$, we can repeat it with independent randomness and amplify its advantage using a majority vote to obtain an error probability of at most 1/3. By applying a Chernoff bound and using Theorem 2, we can lower bound the communication cost required to achieve advantage of γ by $\Omega(\gamma^2 N)$. Unfortunately, this bound is insufficient for our purpose of obtaining a tight streaming switching lemma. On the other hand, relatively recent results [6, 13] prove a much stronger lower bound of $\Omega(\gamma N)$ on the communication cost by a more careful analysis. This stronger bound (summarized in the theorem below) will allow us to prove a tight streaming switching lemma. Nevertheless, we use the full power of the theorem only in the multi-pass version of the lemma in Section 5, whereas the main (single-pass) lemma only requires a weaker variant of the theorem for one-way communication protocols.

Theorem 3 (unique-disjointness bound). *There exists a constant $M \geq 1$ for which any public-coin randomized protocol Π for unique-disjointness that satisfies $\mathrm{Adv}_N^{\mathrm{UDISJ}}(\Pi) = \gamma$ must communicate at least $\frac{1}{M}\gamma N - M \log N$ bits in the worst case.*

The proof is heavily based on the proof of Theorem 2.2 in [6]. It is described in Appendix A for the sake of completeness, where we prove it with $M = 20$.

[6] Our notation for disjointness is consistent with the rest of the paper, yet it differs from standard notation used in communication complexity.

4 The Streaming Switching Lemma

Our main theorem is stated below. We refer to it as a "streaming switching lemma" (for the sake of compatibility with previous results).

Theorem 4 (streaming switching lemma). *There exists a constant $M_1 \geq 1$ such that any S-bounded randomized algorithm $\mathcal{A}$ for $S \geq \log N$ with access to a stream containing $\log N \leq Q \leq N/3$ elements drawn from $[N]$ via either a random permutation or a random function has a distinguishing advantage bounded by*

$$\mathrm{Adv}^{\mathrm{STR}}_{\mathcal{D}_N^Q, \mathcal{D}_N^{1\times Q}}(\mathcal{A}) \leq \mathrm{Opt}^{\mathrm{STR}}_{\mathcal{D}_N^Q, \mathcal{D}_N^{1\times Q}}(S) \leq \frac{M_1 \cdot \lceil \log Q \rceil \cdot Q}{N} \cdot (S + M_1 \cdot \log N).$$

Remark 1. The advantage is $O(\log Q \cdot Q \cdot S/N)$ given than $S = \Omega(\log N)$.

Remark 2. It follows from our proof that we can set $M_1 = 30$. However, a smaller value of M_1 can be derived by low-level optimizations.

Theorem 4 follows from the lemma below, which is proved in Section 4.1.

Lemma 1. *There exists a constant $M_1 \geq 1$ such that for any $K \leq N/3$ and $S \geq \log N$,*

$$\mathrm{Opt}^{\mathrm{STR}}_{\mathcal{D}_N^{2K}, \mathcal{D}_N^{K\times 2}}(S) \leq \frac{M_1 \cdot K}{N} \cdot (S + M_1 \cdot \log N).$$

Proof (of Theorem 4). Let M_1 be the constant implied by Lemma 1. We denote by $\Gamma = \Gamma(N, S) = \frac{M_1}{N} \cdot (S + M_1 \cdot \log N)$ the upper bound on $\mathrm{Opt}^{\mathrm{STR}}_{\mathcal{D}_N^{2K}, \mathcal{D}_N^{K\times 2}}(S)$ deduced in Lemma 1, divided by K. Note that $\Gamma(N, S)$ does not depend on K. Let k be a positive integer such that $K = 2^k < 2N/3$. We prove that for any S-bounded algorithm $\mathcal{A}$ with $S \geq \log N$,

$$\mathrm{Adv}^{\mathrm{STR}}_{\mathcal{D}_N^K, \mathcal{D}_N^{1\times K}}(\mathcal{A}) \leq \frac{k \cdot K}{2} \cdot \Gamma. \tag{1}$$

The proof is by induction on k. The base case is for k such that $K \leq \log N$. It follows from the original switching lemma (Theorem 1), since

$$\mathrm{Adv}^{\mathrm{STR}}_{\mathcal{D}_N^K, \mathcal{D}_N^{1\times K}}(\mathcal{A}) \leq \frac{K^2}{2N} \leq \frac{K \cdot S}{2N} \leq \frac{M_1 \cdot k \cdot K}{2N} \cdot (S + M_1 \cdot \log N).$$

Suppose that the hypothesis holds up to $k' = k$. We prove it for $k' = k + 1$ (assuming $K \leq N/3$). We have

$$\begin{aligned}
\mathrm{Adv}^{\mathrm{STR}}_{\mathcal{D}_N^{2K},\mathcal{D}_N^{1\times 2K}}(\mathcal{A}) &= \\
\left|\Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{2K})} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{1\times 2K})} \Rightarrow 1]\right| &= \\
\Big|\Big(\Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{2K})} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{K\times 2})} \Rightarrow 1]\Big) &+ \\
\Big(\Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{K\times 2})} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{1\times 2K})} \Rightarrow 1]\Big)\Big| &\leq \\
\Big|\Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{2K})} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{K\times 2})} \Rightarrow 1]\Big| &+ \\
\Big|\Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{K\times 2})} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{1\times 2K})} \Rightarrow 1]\Big| &\leq \qquad\qquad \text{(Lemma 1)} \\
K \cdot \mathit{\Gamma} &+ \\
\Big|\Big(\Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{K\times 2})} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{K}\|\mathcal{D}_N^{1\times K})} \Rightarrow 1]\Big) &+ \\
\Big(\Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{K}\|\mathcal{D}_N^{1\times K})} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{1\times 2K})} \Rightarrow 1]\Big)\Big| &\leq \\
K \cdot \mathit{\Gamma} &+ \\
\Big|\Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{K}\|\mathcal{D}_N^{K})} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{K}\|\mathcal{D}_N^{1\times K})} \Rightarrow 1]\Big| &+ \\
\Big|\Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{K}\|\mathcal{D}_N^{1\times K})} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathrm{O}(\mathcal{D}_N^{1\times K}\|\mathcal{D}_N^{1\times K})} \Rightarrow 1]\Big| &\leq \qquad\qquad \text{(hypothesis)} \\
K \cdot \mathit{\Gamma} + 2 \cdot \frac{k \cdot K}{2} \cdot \mathit{\Gamma} &= \\
\frac{(k+1) \cdot 2K}{2} \cdot \mathit{\Gamma}&.
\end{aligned}$$

This completes the proof of the induction.

Finally, let $\mathcal{A}$ be S-bounded as in the theorem. Let $q' = \lceil \log Q \rceil$ and $Q' = 2^{q'}$ (note that $Q \leq Q' \leq 2Q$). We have

$$\mathrm{Adv}^{\mathrm{STR}}_{\mathcal{D}_N^{Q},\mathcal{D}_N^{1\times Q}}(\mathcal{A}) \leq \mathrm{Adv}^{\mathrm{STR}}_{\mathcal{D}_N^{Q'},\mathcal{D}_N^{1\times Q'}}(\mathcal{A}) \leq \frac{q' \cdot Q'}{2} \cdot \mathit{\Gamma} \leq \lceil \log Q \rceil \cdot Q \cdot \mathit{\Gamma},$$

where the second inequality follows from (1). This concludes the proof of Theorem 4. ■

4.1 Reduction from Communication Complexity to Streaming

We now define the permutation-dependence problem and summarize the outcome of the reduction from this problem to streaming in Proposition 1. We then state a lower bound on the communication cost of the permutation-dependence problem in Proposition 2 (which is proved in Section 4.2), and use it to prove Lemma 1.

The Permutation-Dependence Problem *Permutation-dependence* is a 2-player game between $\mathcal{A}$ and $\mathcal{B}$ that run a protocol Π. For an even parameter $K \leq N$, we choose the K elements

$$x_1, \ldots, x_{K/2}, y_1, \ldots, y_{K/2},$$

from either $\mathcal{D}_N^K$, or from $\mathcal{D}_N^{K/2 \times 2}$. We give $x_1, \ldots, x_{K/2}$ to $\mathcal{A}$ and $y_1, \ldots, y_{K/2}$ to $\mathcal{B}$. Note that regardless of the distribution from which the K elements are chosen, the input to each player is taken from the (marginal) distribution $\mathcal{D}_N^{K/2}$. However, the inputs are either dependent (chosen from $\mathcal{D}_N^K$) or independent (chosen from $\mathcal{D}_N^{K/2 \times 2}$) and the goal of the players is to distinguish between these cases.

After receiving their inputs x, y, players $\mathcal{A}, \mathcal{B}$ run a communication protocol Π and then one of the players outputs a bit which is the output of the protocol, denoted by $PDEP_\Pi(x, y)$. We say that Π has communication cost C if $\mathcal{A}, \mathcal{B}$ communicate at most C bits in all possible protocol executions. Similarly to the disjointness problem, we will be interested in public-coin randomized protocols for permutation-dependence.

Since it is a distributional communication complexity problem, we define the following notation for permutation-dependence:

$$\begin{aligned}
\mathrm{Err}_{N,K}^{\mathrm{PDEP0}}(\Pi) &= \Pr[PDEP_\Pi(x, y) \neq 0 \mid x, y \overset{\$}{\leftarrow} \mathcal{D}_N^{K/2 \times 2}], \\
\mathrm{Err}_{N,K}^{\mathrm{PDEP1}}(\Pi) &= \Pr[PDEP_\Pi(x, y) \neq 1 \mid x, y \overset{\$}{\leftarrow} \mathcal{D}_N^{K}], \\
\mathrm{Adv}_{N,K}^{\mathrm{PDEP}}(\Pi) &= \left|1 - \mathrm{Err}_{N,K}^{\mathrm{PDEP1}}(\Pi) - \mathrm{Err}_{N,K}^{\mathrm{PDEP0}}(\Pi)\right|, \\
\mathrm{Opt}_{N,K}^{\mathrm{PDEP}}(C) &= \max_{\Pi}\{\ \mathrm{Adv}_{N,K}^{\mathrm{PDEP}}(\Pi) \mid \Pi \text{ has communication cost } C\}.
\end{aligned}$$

We further denote by $\mathrm{Opt}_{N,K}^{\mathrm{PDEP}\rightarrow}(C)$ the optimal advantage of a *one-way communication protocol* for permutation-dependence. Namely, we only consider protocols in which $\mathcal{A}$ sends a single message to $\mathcal{B}$, which outputs the answer. Clearly, $\mathrm{Opt}_{N,K}^{\mathrm{PDEP}\rightarrow}(C) \leq \mathrm{Opt}_{N,K}^{\mathrm{PDEP}}(C)$.

The Reduction from Permutation-Dependence to Streaming The following proposition upper bounds the advantage of a (memory-bounded) streaming algorithm in distinguishing between $\mathcal{D}_N^K$ and $\mathcal{D}_N^{K/2 \times 2}$ by the advantage of an optimal one-way permutation-dependence protocol (with limited communication cost). It is a standard reduction from a 2-player one-way communication protocol to streaming (for example, refer to [19]).

Proposition 1. *For any S and even $K \leq N$,*

$$\mathrm{Opt}_{\mathcal{D}_N^K, \mathcal{D}_N^{K/2 \times 2}}^{\mathrm{STR}}(S) \leq \mathrm{Opt}_{N,K}^{\mathrm{PDEP}\rightarrow}(S).$$

Proof. Given black-box access to an S-bounded streaming algorithm $\mathcal{A}_1$, players $\mathcal{A}$ and $\mathcal{B}$ in the permutation-dependence protocol Π run $\mathcal{A}_1$ and answer its oracle queries using their inputs: $\mathcal{A}$ answers the first batch of $K/2$ queries (using $x_1, \ldots, x_{K/2}$) and then communicates the intermediate state of $\mathcal{A}_1$ to $\mathcal{B}$ which answers the second batch of $K/2$ queries (using $y_1, \ldots, y_{K/2}$). Finally, $\mathcal{B}$ outputs the same answer as $\mathcal{A}_1$.

Thus, $\mathcal{A}_1$ is given oracle access to O, where either $\mathrm{O} = \mathrm{O}(\mathcal{D}_N^K)$ or $\mathrm{O} = \mathrm{O}(\mathcal{D}_N^{K/2\times 2})$, depending on the distribution of the inputs x, y of $\mathcal{A}, \mathcal{B}$. Clearly, Π is a one-way communication protocol. Moreover, since $\mathcal{A}_1$ is S-bounded and its state is communicated once, the communication cost of Π is bounded by S. Therefore,

$$\mathrm{Adv}^{\mathrm{STR}}_{\mathcal{D}_N^K, \mathcal{D}_N^{K/2\times 2}}(\mathcal{A}_1) = \mathrm{Adv}^{\mathrm{PDEP}}_{N,K}(\Pi) \leq \mathrm{Opt}^{\mathrm{PDEP}\rightarrow}_{N,K}(S).$$

The proposition follows since the above inequality holds for any S-bounded algorithm $\mathcal{A}_1$. ■

Remark 3. In case $S > K/2$, a trivial reduction (where one party sends its input to the other) is more efficient than the one above. This gives

$$\mathrm{Opt}^{\mathrm{STR}}_{\mathcal{D}_N^K, \mathcal{D}_N^{K/2\times 2}}(S) \leq \mathrm{Opt}^{\mathrm{PDEP}\rightarrow}_{N,K}(K/2).$$

Using this observation, it is possible to obtain a limited improvement to the streaming switching lemma (Theorem 4) in case $S = N^{\Omega(1)}$.

Proof of Lemma 1 In order to prove Lemma 1, we use the following proposition (proved in Section 4.2) which bounds the advantage of any protocol Π for permutation-dependence.

Proposition 2. *There exists a constant $M_1 \geq 1$ such that for any $K \leq N/3$ and $C \geq \log N$,*

$$\mathrm{Opt}^{\mathrm{PDEP}}_{N,2K}(C) \leq \frac{M_1 \cdot K}{N} \cdot (C + M_1 \cdot \log N).$$

Proof (of Lemma 1). Let M_1 be the constant implied by Proposition 2. Based on Proposition 1 and Proposition 2 we have

$$\mathrm{Opt}^{\mathrm{STR}}_{\mathcal{D}_N^{2K}, \mathcal{D}_N^{K\times 2}}(S) \leq \mathrm{Opt}^{\mathrm{PDEP}\rightarrow}_{N,2K}(S) \leq \mathrm{Opt}^{\mathrm{PDEP}}_{N,2K}(S) \leq \frac{M_1 \cdot K}{N} \cdot (S + M_1 \cdot \log N).$$

■

Remark 4. Proposition 2 upper bounds $\mathrm{Opt}^{\mathrm{PDEP}}_{N,2K}(C)$, yet the proof of Lemma 1 only requires an upper bound on $\mathrm{Opt}^{\mathrm{PDEP}\rightarrow}_{N,2K}(S)$. This suggests that a (small) improvement to the bound of Lemma 1 (and hence to the bound of Theorem 4) may be possible.

4.2 Reduction from Unique-Disjointness to Permutation-Dependence

The proof of Proposition 2 is based on a reduction from the unique-disjointness problem to the permutation-dependence problem, summarized by the proposition below.

Proposition 3. *Let* $K \leq N/3$ *and* $N' = \lfloor N/K \rfloor$. *There exists a public-coin randomized local reduction,* f_1, f_2, *where* $f_i : \{0,1\}^{N'} \to [N]^K$, *such that for any* $a, b \in \{0,1\}^{N'} \times \{0,1\}^{N'}$,

$$f_1(a), f_2(b) \sim \begin{cases} \mathcal{D}_N^{K \times 2}, & \textit{if } UDISJ(a,b) = 0 \\ \mathcal{D}_N^{2K}, & \textit{if } UDISJ(a,b) = 1. \end{cases}$$

Here, a public-coin randomized local reduction means that f_1 only depends on a and on public randomness (but not on b), and similarly, f_2 does not depend on a. Hence, if a, b intersect at exactly 1 index, then the output of the reduction consists of two independent random permutation streams, each of K elements. On the other hand, if a, b are disjoint, then the output of the reduction consists of a single random permutation stream of $2K$ elements (that is split into two halves).

Proof. We describe the reduction f_1, f_2 as a procedure executed by two parties $\mathcal{A}, \mathcal{B}$ that do not communicate, but share a random string.

1. Given incidence vector inputs (bit arrays) $a, b \in \{0,1\}^{N'} \times \{0,1\}^{N'}$, let $S_A = a^{(K)} \| 0^{(N - N' \cdot K)}, S_B = b^{(K)} \| 0^{(N - N' \cdot K)}$. Namely, each party locally duplicates its array K times and appends zero entries such that $S_A \in \{0,1\}^N$ and $S_B \in \{0,1\}^N$.
2. Using their joint randomness, the parties sample a sequence of K indices $i_1, i_2, \ldots, i_K \xleftarrow{\$} \mathcal{D}_N^K$ (chosen from $[N]$ without replacement). The parties use the sampled indices to create new arrays: $\mathcal{A}$ defines an array $T_A \in \{0,1\}^K$, where $T_A[j] = S_A[i_j]$ for $j \in \{1, 2, \ldots, K\}$. Similarly, $\mathcal{B}$ defines $T_B \in \{0,1\}^K$, where $T_B[j] = S_B[i_j]$ for $j \in \{1, 2, \ldots, K\}$.
3. Each party locally extends its array from size K to size N such that its Hamming weight becomes K (the parties add disjoint 1 entries). More specifically, $\mathcal{A}$ computes

$$T_A^2 = T_A \| 1^{(K - HW(T_A))} \| 0^{(N - 2K + HW(T_A))},$$

   and $\mathcal{B}$ computes

$$T_B^2 = T_B \| 0^{(K)} \| 1^{(K - HW(T_B))} \| 0^{(N - 3K + HW(T_B))}.$$

4. Each party applies (the same) uniform permutation $\sigma : \{0,1\}^N \to \{0,1\}^N$ to its array of size N (σ is specified in the joint randomness),

$$T_A^3[i] = T_A^2[\sigma(i)], \text{ and } T_B^3[i] = T_B^2[\sigma(i)],$$

for each $i \in [N]$.

5. Finally, $\mathcal{A}$ selects a uniform permutation $\sigma_1 : \{0,1\}^K \to \{0,1\}^K$ and uses it to output the elements indicated by its array T_A^3 (the 1 entries) in uniform order. $\mathcal{A}$ outputs

$$f_1(a)_i = SEQ(T_A^3)_{\sigma_1(i)}, \text{ for each } i \in [K].$$

$\mathcal{B}$ selects a uniform permutation $\sigma_2 : \{0,1\}^K \to \{0,1\}^K$ and outputs

$$f_2(b)_i = SEQ(T_B^3)_{\sigma_2(i)}, \text{ for each } i \in [K].$$

Analysis Observe that $T_A^3 \in \{0,1\}^N$ satisfies $HW(T_A^3) = K$ and similarly $T_B^3 \in \{0,1\}^N$ satisfies $HW(T_B^3) = K$. Therefore, each party outputs a sequence of K elements.

Due to the randomization of σ (which randomizes the elements that are output by f_1, f_2) and of σ_1, σ_2 (which randomize the order of the elements output by f_1, f_2), we have the following property.

Property 1. Let $a, b \in \{0,1\}^{N'} \times \{0,1\}^{N'}$ and

$$x, y = x_1, \ldots, x_K, y_1, \ldots, y_K \in [N]^{2K},\ x', y' = x'_1, \ldots, x'_K, y'_1, \ldots, y'_K \in [N]^{2K},$$

where each K element sequence (x, y, x' and y') contains distinct elements and for some $0 \leq t \leq K$,

$$|\{x_1, \ldots, x_K\} \cap \{y_1, \ldots, y_K\}| = |\{x'_1, \ldots, x'_K\} \cap \{y'_1, \ldots, y'_K\}| = t.$$

Then,

$$\Pr[f_1(a), f_2(b) = x, y] = \Pr[f_1(a), f_2(b) = x', y'].$$

Hence, the distribution of $f_1(a), f_2(b)$ is completely determined by the distribution of the size of the intersection of the sequences $f_1(a)$ and $f_2(b)$ as sets. The intersection size is equal to $|T_A \cap T_B|$ (since $|T_A \cap T_B| = |T_A^3 \cap T_B^3|$), thus we analyze this variable below.

Observe that

$$|S_A \cap S_B| = K \cdot |a \cap b|.$$

Consider the case that $UDISJ(a, b) = 1$, or $|a \cap b| = 0$. We have $|S_A \cap S_B| = 0$ and therefore $|T_A \cap T_B| = 0$. Hence, $f_1(a)$ and $f_2(b)$ are disjoint as sets, and by Property 1, $f_1(a), f_2(b) \sim \mathcal{D}_N^{2K \times 1}$.

Otherwise, $UDISJ(a, b) = 0$, implying that $|a \cap b| = 1$ and therefore $|S_A \cap S_B| = K$. The number of options for selecting $i_1, i_2, \ldots, i_K$ in the second step such that they intersect the K common indices in S_A, S_B in exactly $0 \leq t \leq K$ places is $\binom{K}{t}\binom{N-K}{K-t}$. Since the total number of options for selecting $i_1, i_2, \ldots, i_K$ is $\binom{N}{K}$,

$$\Pr[|T_A \cap T_B| = t] = \frac{\binom{K}{t}\binom{N-K}{K-t}}{\binom{N}{K}}.$$

At the same time,

$$\Pr\left[\left|\{x_1,\dots,x_K\} \cap \{y_1,\dots,y_K\}\right| = t \mid x_1,\dots,x_K,y_1,\dots,y_K \xleftarrow{\$} \mathcal{D}_N^{K\times 2}\right] =$$

$$\frac{\binom{K}{t}\binom{N-K}{K-t}}{\binom{N}{K}} = \Pr[|T_A \cap T_B| = t].$$

Hence, by Property 1, $f_1(a), f_2(b) \sim \mathcal{D}_N^{K\times 2}$ as claimed. ■

Finally, Proposition 2 follows from Proposition 3 and Theorem 3.

Proof (of Proposition 2). We show that there exists a constant M_1 such that any permutation-dependence protocol $\mathit{\Pi}'$ with communication cost $C \geq \log N$ satisfies $\mathrm{Adv}_{N,2K}^{\mathrm{PDEP}}(\mathit{\Pi}') \leq \frac{M_1\cdot K}{N} \cdot (C + M_1 \cdot \log N)$. This proves Proposition 2.

Fix a permutation-dependence protocol $\mathit{\Pi}'$ as above. We consider a protocol $\mathit{\Pi}$ for unique-disjointness, where given an input $a, b \in \{0,1\}^{N'} \times \{0,1\}^{N'}$ (for $N' = \lfloor N/K \rfloor$), each party independently applies the reduction of Proposition 3 to its input using the public randomness. The parties then run the permutation-dependence protocol $\mathit{\Pi}'$ on input $f_1(a), f_2(b)$ with communication cost (at most) C bits in the worst case and output the same value. In short,

$$UDISJ_{\mathit{\Pi}}(a,b) = PDEP_{\mathit{\Pi}'}(f_1(a), f_2(b)).$$

Proposition 3 implies that for every a, b such that $UDISJ(a,b) = 0$,

$$\Pr[UDISJ_{\mathit{\Pi}}(a,b) = 1 \mid UDISJ(a,b) = 0] =$$

$$\Pr[PDEP_{\mathit{\Pi}'}(f_1(a), f_2(b)) = 1 \mid UDISJ(a,b) = 0] = \mathrm{Err}_{N,2K}^{\mathrm{PDEP0}}(\mathit{\Pi}'),$$

and a similar equality holds for every a, b such that $UDISJ(a,b) = 1$. Hence

$$\mathrm{Err}_{N'}^{\mathrm{UDISJ0}}(\mathit{\Pi}) = \mathrm{Err}_{N,2K}^{\mathrm{PDEP0}}(\mathit{\Pi}'), \text{ and } \mathrm{Err}_{N'}^{\mathrm{UDISJ1}}(\mathit{\Pi}) = \mathrm{Err}_{N,2K}^{\mathrm{PDEP1}}(\mathit{\Pi}').$$

Denote

$$\alpha' = 1 - \mathrm{Err}_{N'}^{\mathrm{UDISJ1}}(\mathit{\Pi}),\ \beta' = \mathrm{Err}_{N'}^{\mathrm{UDISJ0}}(\mathit{\Pi}),$$

and $\gamma' = \alpha' - \beta'$. We have

$$\mathrm{Adv}_{N'}^{\mathrm{UDISJ}}(\mathit{\Pi}) = \alpha' - \beta' = \gamma' =$$

$$1 - \mathrm{Err}_{N,2K}^{\mathrm{PDEP1}}(\mathit{\Pi}') - \mathrm{Err}_{N,2K}^{\mathrm{PDEP0}}(\mathit{\Pi}') = \mathrm{Adv}_{N,2K}^{\mathrm{PDEP}}(\mathit{\Pi}'),$$

where we assume that $\alpha' - \beta' \geq 0$ (otherwise, $\mathcal{A}, \mathcal{B}$ in $\mathit{\Pi}$ simply negate the output of $\mathit{\Pi}'$). Hence, γ' is equal to the advantage of both the unique-disjointness and permutation-dependence protocols.

We apply Theorem 3 to $\mathit{\Pi}$, and since C upper bounds the communication cost of $\mathit{\Pi}$ in the worst case, we conclude that $C \geq \frac{1}{M} \cdot N' \cdot \gamma' - M \log N'$. This gives

$$\gamma' \leq \frac{M}{N'} \cdot (C + M \cdot \log N') \leq \frac{M}{N'} \cdot (C + M \cdot \log N).$$

Define $M_1 = 3/2 \cdot M$. Note that since $K \leq N/3$, then

$$N' = \left\lfloor \frac{N}{K} \right\rfloor \geq \frac{N-K}{K} \geq \frac{2N}{3K},$$

hence $\frac{M}{N'} \leq \frac{M_1 \cdot K}{N}$. Therefore,

$$\gamma' \leq \frac{M_1 \cdot K}{N} \cdot (C + M_1 \cdot \log N),$$

as claimed. ■

5 The Multi-Pass Streaming Switching Lemma

For a parameter $P \geq 1$, we consider a P-pass streaming algorithm which can access an input stream of Q elements P times at the same order. The P-pass algorithm attempts to distinguish between a stream chosen from a random permutation or from a random function. In our model, the algorithm interacts with an oracle that samples from one of the distributions defined below.

For $0 < K \leq N$, let $\mathcal{D}_N^{K \times R \otimes P}$ be the distribution over $[N]^{K \cdot R \cdot P}$ that is defined by a sampling procedure which first draws $x \xleftarrow{\$} \mathcal{D}_N^{K \times R}$ and then outputs $\underbrace{x \| x \| \ldots \| x}_{P}$. In case $R = 1$, we simply write $\mathcal{D}_N^{K \otimes P}$.

Theorem 5 (multi-pass switching lemma). *There exists a constant $M_1 \geq 1$ such that any S-bounded randomized P-pass algorithm $\mathcal{A}$ for $S \geq \log N$ with access to a stream containing $\log N \leq Q \leq N/3$ elements drawn from $[N]$ via either a random permutation or a random function has a distinguishing advantage bounded by*

$$\mathrm{Adv}^{\mathrm{STR}}_{\mathcal{D}_N^{Q \otimes P}, \mathcal{D}_N^{1 \times Q \otimes P}}(\mathcal{A}) \leq \mathrm{Opt}^{\mathrm{STR}}_{\mathcal{D}_N^{Q \otimes P}, \mathcal{D}_N^{1 \times Q \otimes P}}(S) \leq \frac{M_1 \cdot \lceil \log Q \rceil \cdot Q}{N} \cdot (P \cdot S + M_1 \cdot \log N).$$

The proof of Theorem 5 is based on the lemma below, which is a generalization of Lemma 1.

Lemma 2. *There exists a constant $M_1 \geq 1$ such that for any $K \leq N/3$ and $S \geq \log N$,*

$$\mathrm{Opt}^{\mathrm{STR}}_{\mathcal{D}_N^{2K \otimes P}, \mathcal{D}_N^{K \times 2 \otimes P}}(S) \leq \frac{M_1 \cdot K}{N} \cdot (P \cdot S + M_1 \cdot \log N).$$

We omit the proof of Theorem 5, as it is essentially identical to the one of Theorem 4.

The proof of Lemma 2 uses the following proposition which generalizes Proposition 1.

Proposition 4. *For any S and even $K \leq N$,*

$$\mathrm{Opt}^{\mathrm{STR}}_{\mathcal{D}_N^{K \otimes P}, \mathcal{D}_N^{K/2 \times 2 \otimes P}}(S) \leq \mathrm{Opt}^{\mathrm{PDEP}}_{N,K}(P \cdot S).$$

Proof. The proof is via a reduction from the (multi-round) permutation-dependence problem to (multi-pass) streaming, which generalizes the proof of Proposition 1. The only difference is that in order to simulate the P-pass streaming algorithm, its state is communicated P times between the parties, hence the communication cost of the permutation-dependence protocol is bounded by $S \cdot P$. ■

Proof (of Lemma 2). Let M_1 be the constant implied by Proposition 2. Based on Proposition 4 and Proposition 2 we have

$$\mathrm{Opt}^{\mathrm{STR}}_{\mathcal{D}_N^{2K \otimes P}, \mathcal{D}_N^{K \times 2 \otimes P}}(S) \leq \mathrm{Opt}^{\mathrm{PDEP}}_{N,2K}(P \cdot S) \leq \frac{M_1 \cdot K}{N} \cdot (P \cdot S + M_1 \cdot \log N).$$

■

6 Conclusions and Future Work

In this paper we proved an upper bound on the streaming distinguishing advantage between a random permutation and a random function, which is tight up to poly-logarithmic factors. Our proof is based on a hybrid argument that gives rise to a reduction from the unique-disjointness communication complexity problem to streaming. In the future, it would be interesting to apply our techniques to additional streaming problems that are relevant to cryptography.

Acknowledgements The author would like to thank Andrej Bogdanov for his helpful comment on a previous version of this work, which allowed to base the single-pass streaming switching lemma on a permutation-dependence problem with one-way communication (the previous version was based on a generalized variant of permutation-dependence with multi-round communication).

The author was supported by the Israeli Science Foundation through grant No. 573/16 and by the European Research Council under the ERC starting grant agreement No. 757731 (LightCrypt).

References

1. N. Alon, Y. Matias, and M. Szegedy. The Space Complexity of Approximating the Frequency Moments. *J. Comput. Syst. Sci.*, 58(1):137–147, 1999.
2. Z. Bar-Yossef, T. S. Jayram, R. Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *J. Comput. Syst. Sci.*, 68(4):702–732, 2004.
3. M. Bellare and P. Rogaway. The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs. In S. Vaudenay, editor, *Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28 - June 1, 2006, Proceedings*, volume 4004 of *Lecture Notes in Computer Science*, pages 409–426. Springer, 2006.

4. K. Bhargavan and G. Leurent. On the Practical (In-)Security of 64-bit Block Ciphers: Collision Attacks on HTTP over TLS and OpenVPN. In E. R. Weippl, S. Katzenbeisser, C. Kruegel, A. C. Myers, and S. Halevi, editors, *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, pages 456–467. ACM, 2016.
5. M. Braverman and A. Moitra. An Information Complexity Approach to Extended Formulations. *Electronic Colloquium on Computational Complexity (ECCC)*, 19:131, 2012.
6. M. Braverman and A. Moitra. An information complexity approach to extended formulations. In D. Boneh, T. Roughgarden, and J. Feigenbaum, editors, *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 161–170. ACM, 2013.
7. R. P. Brent. An improved Monte Carlo factorization algorithm. *BIT Numerical Mathematics*, 20(2):176–184, 1980.
8. C. Cachin and U. M. Maurer. Unconditional Security Against Memory-Bounded Adversaries. In B. S. K. Jr., editor, *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings*, volume 1294 of *Lecture Notes in Computer Science*, pages 292–306. Springer, 1997.
9. A. Chattopadhyay and T. Pitassi. The story of set disjointness. *SIGACT News*, 41(3):59–85, 2010.
10. T. M. Cover and J. A. Thomas. *Elements of information theory (2. ed.)*. Wiley, 2006.
11. W. Dai, V. T. Hoang, and S. Tessaro. Information-Theoretic Indistinguishability via the Chi-Squared Method. In J. Katz and H. Shacham, editors, *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III*, volume 10403 of *Lecture Notes in Computer Science*, pages 497–523. Springer, 2017.
12. O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *J. ACM*, 33(4):792–807, 1986.
13. M. Göös and T. Watson. Communication Complexity of Set-Disjointness for All Probabilities. *Theory of Computing*, 12(1):1–23, 2016.
14. C. Hall, D. A. Wagner, J. Kelsey, and B. Schneier. Building PRFs from PRPs. In H. Krawczyk, editor, *Advances in Cryptology - CRYPTO '98, 18th Annual International Cryptology Conference, Santa Barbara, California, USA, August 23-27, 1998, Proceedings*, volume 1462 of *Lecture Notes in Computer Science*, pages 370–389. Springer, 1998.
15. R. Impagliazzo and S. Rudich. Limits on the Provable Consequences of One-Way Permutations. In D. S. Johnson, editor, *Proceedings of the 21st Annual ACM Symposium on Theory of Computing, May 14-17, 1989, Seattle, Washigton, USA*, pages 44–61. ACM, 1989.
16. J. Jaeger and S. Tessaro. Tight Time-Memory Trade-Offs for Symmetric Encryption. In Y. Ishai and V. Rijmen, editors, *Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I*, volume 11476 of *Lecture Notes in Computer Science*, pages 467–497. Springer, 2019.
17. B. Kalyanasundaram and G. Schnitger. The Probabilistic Communication Complexity of Set Intersection. *SIAM J. Discrete Math.*, 5(4):545–557, 1992.
18. D. E. Knuth. *The Art of Computer Programming, Volume II: Seminumerical Algorithms*. Addison-Wesley, 1969.

19. E. Kushilevitz and N. Nisan. *Communication complexity.* Cambridge University Press, 1997.
20. I. Newman. Private vs. Common Random Bits in Communication Complexity. *Inf. Process. Lett.*, 39(2):67–71, 1991.
21. N. Nisan. Pseudorandom generators for space-bounded computation. *Combinatorica*, 12(4):449–461, 1992.
22. J. M. Pollard. A monte carlo method for factorization. *BIT Numerical Mathematics*, 15(3):331–334, 1975.
23. A. A. Razborov. On the Distributional Complexity of Disjointness. *Theor. Comput. Sci.*, 106(2):385–390, 1992.
24. S. Tessaro and A. Thiruvengadam. Provable Time-Memory Trade-Offs: Symmetric Cryptography Against Memory-Bounded Adversaries. In *Theory of Cryptography - 16th International Conference, TCC 2018, Panaji, India, November 11-14, 2018, Proceedings, Part I*, pages 3–32, 2018.
25. T. Watson. Communication Complexity with Small Advantage. In R. A. Servedio, editor, *33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA*, volume 102 of *LIPIcs*, pages 9:1–9:17. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018.

A Concrete Parameters for Theorem 3

In this appendix we prove Theorem 3 for $M = 20$, as restated below.

Theorem 3 (restated with $M = 20$). *Any public-coin randomized protocol Π for unique-disjointness that satisfies $\mathrm{Adv}_N^{\mathrm{UDISJ}}(\Pi) = \gamma$ must communicate at least $\frac{1}{20}\gamma N - 20\log N$ bits in the worst case.*

We first describe information theory preliminaries, which are heavily used in the proof (for more details refer to [10]). We then give an overview of the proof, which is based on the proof of Theorem 2.2 in [5, revision 1] (an updated version of [6]).

A.1 Information Theory

We begin with notations and definitions. Consider discrete random variables X, Y, Z. We denote the distribution of X by $p(X)$. We denote by $\mathcal{X}(x)$ the probability that a random variable drawn from the distribution $\mathcal{X}$ gets the value x.

The *entropy* of X is

$$\mathrm{H}(X) = \sum_x \Pr[X = x] \log(1/\Pr[X = x]).$$

The *conditional entropy* of X given Y is

$$\mathrm{H}(X|Y) = \sum_y \Pr[Y = x] H(X|Y = y) = \mathrm{H}[X, Y] - \mathrm{H}[Y].$$

The *mutual information* between X, Y is

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X),$$

where $I(X;Y) = 0$ if and only if X and Y are independent. The *conditional mutual information* between X, Y given Z is

$$I(X;Y|Z) = H(X|Z) - H(X|Y,Z).$$

The *Kullback-Leibler divergence* (also known as the relative entropy) between two distributions $\mathcal{X}, \mathcal{Y}$ is

$$\mathrm{D}(\mathcal{X}\|\mathcal{Y}) = \sum_x \mathcal{X}(x)\log(\mathcal{X}(x)/\mathcal{Y}(x)).$$

Next, we describe the properties that we use.

The *chain rule of mutual information* asserts that

$$I(X;Y,Z) = I(X;Z) + I(X;Y|Z).$$

Since (conditional) mutual information is non-negative, this implies that

$$I(X;Y,Z) \geq I(X;Z).$$

We will use the following equalities:

$$\begin{aligned} I(X;Y) =& \sum_x \Pr[X = x]\mathrm{D}(p(Y|X=x)\|p(Y)), \text{ and} \\ I(X;Y|Z) =& \sum_z \Pr[Z = z]\mathrm{D}(p(X,Y|Z=z)\|p(X|Z=z), p(Y|Z=z)) = \\ & \sum_{y,z} \Pr[Y = y, Z = x]\mathrm{D}(p(X|Y=y,Z=z)\|p(X|Z=z)). \end{aligned}$$

Finally, *Pinsker's inequality* bounds the statistical distance between probability distributions as

$$\mathit{\Delta}(\mathcal{X},\mathcal{Y}) \leq \sqrt{1/2 \cdot \mathrm{D}(\mathcal{X}\|\mathcal{Y})}.$$

A.2 Overview of the Proof

The main part of the proof (described in Section A.3) establishes a similar result to Theorem 3 for private-coin protocols. It is based on the proof of Theorem 2.2 by Braverman and Moitra [5, revision 1]. Then, in Section A.4, we complete the proof of the theorem by extending the result to public-coin protocols using the standard sparsification technique of [20].

We now give a short overview of the lower bound proof for private-coin protocols. It uses the *information complexity* approach, which has become a standard technique for proving communication complexity lower bounds (cf., [2]). In particular, we define a distribution on the inputs of the parties, which become random variables, denoted by (A, B). We analyze the amount of information that the concatenation of the messages in the protocol (namely, the protocol transcript, denoted as $\Pi(A, B)$) reveals to each player about the other player's input. This quantity is exactly $\mathrm{I}(A; \Pi \mid B) + \mathrm{I}(B; \Pi \mid A)$ (known as the internal information complexity of Π) and it immediately lower bounds $\mathrm{H}(\Pi)$ and hence the communication cost of Π.

In order to lower bound $\mathrm{I}(A; \Pi \mid B) + \mathrm{I}(B; \Pi \mid A)$, we break down the inputs (A, B) into N mutually independent coordinates (A_j, B_j). This allows using a direct sum property which reduces the task of proving an $\Omega(\epsilon N)$ lower bound for the original problem (for $0 < \epsilon \leq 1$) to the task of proving an $\Omega(\epsilon)$ lower bound for a small "gadget". In particular, the disjointness function can be written as $DISJ(a, b) = \neg \vee_{j \in [N]} (a_j \wedge b_j)$. Hence, in standard proofs that use this approach the gadget is the AND (or NAND) gate.

Unfortunately, it is shown in [5] that there is a protocol for AND that achieves an advantage of γ, but reveals only $O(\gamma^2)$ bits of information. This implies that the standard reduction to the AND gate only allows to prove a lower bound of $\Omega(\gamma^2 N)$ on the communication cost (which can also be obtained by straightforward majority amplification, as summarized in Section 3). We note that the protocol of [5] for AND can also be viewed as a one-way communication protocol in which $\mathcal{B}$ outputs the answer. Therefore, the standard reduction to the AND gate does not allow proving the required $\Omega(\gamma N)$ communication cost lower bound even for one-way protocols.

In order to prove a $\Omega(\gamma N)$ lower bound, Braverman and Moitra use a more complex gadget and the main part of the analysis involves proving that any protocol for this gadget that achieves advantage of γ must reveal $\Omega(\gamma)$ bits of information. The analysis essentially breaks the gadget down into 6 smaller AND gadgets which interact in a way that allows proving the required bound.

A.3 A Lower Bound for Private-Coin Protocols

We consider private-coin protocols for unique-disjointness.

Theorem 6. *Any private-coin protocol Π for unique-disjointness that satisfies $\mathrm{Adv}_N^{\mathrm{UDISJ}}(\Pi) = \gamma$ must communicate at least $\frac{1}{19.5}\gamma N$ bits in the worst case.*

Theorem 3 is a concrete variant of Theorem 2.2 in [5, revision 1] and its proof is very similar to that of [5]. However, we present the proof slightly differently and additionally calculate the constants involved. We note that the proof in [5] employed a so-called "smoothing" to the underlying disjointness protocol, yet this not necessary to prove the theorem and hence is omitted.[7]

Consider a private-coin protocol Π such that $\mathrm{Adv}_N^{\mathrm{UDISJ}}(\Pi) = \gamma$. We analyze the information complexity of Π with respect to the following distribution on inputs: we group the N bits into blocks of size exactly three, and for each pair of three bits we generate $a_j, b_j \in \{0, 1\}^3$ (for $j \in [N/3]$) uniformly at random

[7] The fact that "smoothing" is not required is also mentioned in [25, Footnote 7].

from the pairs of strings of length three bits where a_j and b_j have exactly one 1 and two 0's, and a_j and b_j are disjoint. Consequently, there are 6 possible a_j, b_j pairs. We define A, B as random variables for the inputs of the players and $A_j, B_j \in \{1, 2, 3\}$ as random variables for the location of the 1 bit in a_j, b_j, respectively.

We will be interested in lower bounding $\mathrm{H}(\Pi) = \mathrm{H}(\Pi(A, B))$ by proving a lower bound on the internal information complexity $\mathrm{I}(A; \Pi \mid B) + \mathrm{I}(B; \Pi \mid A)$ using the following fact.

Fact 1 $\frac{1}{2}\left[\sum_j \mathrm{I}(A_j; \Pi \mid A_{1\ldots j-1}, B_{j\ldots n}) + \mathrm{I}(B_j; \Pi \mid A_{1\ldots j}, B_{j+1\ldots n})\right] \leq \mathrm{H}[\Pi]$.

Proof. By the chain rule for mutual information we obtain

$$\sum_j \mathrm{I}(A_j; \Pi \mid A_{1\ldots j-1}, B_{j\ldots n}) \leq \sum_j \mathrm{I}(A_j; \Pi, B_{1\ldots j-1} \mid A_{1\ldots j-1}, B_{j\ldots n}) =$$

$$\sum_j \mathrm{I}(A_j; B_{1\ldots j-1} \mid A_{1\ldots j-1}, B_{j\ldots n}) + \sum_j \mathrm{I}(A_j; \Pi \mid A_{1\ldots j-1}, B) =$$

$$\sum_j \mathrm{I}(A_j; \Pi \mid A_{1\ldots j-1}, B) = \mathrm{I}(A; \Pi \mid B) \leq \mathrm{H}(\Pi),$$

where $\mathrm{I}(A_j; B_{1\ldots j-1} \mid A_{1\ldots j-1}, B_{j\ldots n}) = 0$ by independence. Similarly,

$$\sum_j \mathrm{I}(B_j; \Pi \mid A_{1\ldots j}, B_{j+1\ldots n}) \leq \mathrm{H}(\Pi).$$

Therefore,

$$\frac{1}{2}\left[\sum_j \mathrm{I}(A_j; \Pi \mid A_{1\ldots j-1}, B_{j\ldots n}) + \mathrm{I}(B_j; \Pi \mid A_{1\ldots j}, B_{j+1\ldots n})\right] \leq \mathrm{H}(\Pi),$$

concluding the proof. ■

We define $C_j = A_{1\ldots j-1}, B_{j+1\ldots n}$ and write

$$\mathrm{I}(A_j; \Pi \mid A_{1\ldots j-1}, B_{j\ldots n}) = \mathrm{I}(A_j; \Pi \mid C_j, B_j) =$$
$$\sum_{c,i} \sum_t \Pr[\Pi = t, C_j = c, B_j = i] \mathrm{D}(p(A_j \mid \Pi = t, C_j = c, B_j = i) \| p(A_j \mid C_j = c, B_j = i)),$$

and a similar equality holds for $\mathrm{I}(B_j; \Pi \mid A_{1\ldots j}, B_{j+1\ldots n})$.

Choosing C_j according to the distribution on the inputs, we obtain

$$\mathrm{I}(A_j; \Pi \mid C_j, B_j) + \mathrm{I}(B_j; \Pi \mid C_j, A_j) = \sum_t \mathrm{E}[\mathrm{adv}(t, C_j)], \tag{2}$$

where the expectation is over C_j, and $\mathrm{adv}(t, C_j)$ is defined as

$$\mathrm{adv}(t, C_j) =$$
$$\sum_{i \in \{1,2,3\}} \Pr[\Pi = t, B_j = i \mid C_j] \mathrm{D}(p(A_j \mid \Pi = t, B_j = i, C_j) \| p(A_j \mid B_j = i, C_j)) +$$
$$\Pr[\Pi = t, A_j = i \mid C_j] \mathrm{D}(p(B_j \mid \Pi = t, A_j = i, C_j) \| p(B_j \mid A_j = i, C_j)).$$

Since B_j is independent of C_j and is uniform in $\{1,2,3\}$ (and the same property holds for A_j), then

$$\begin{aligned}&\mathrm{adv}(t, C_j) = \\ \frac{1}{3}\sum_{i\in\{1,2,3\}} &\Pr[\mathit{\Pi} = t \mid B_j = i, C_j]\mathrm{D}(p(A_j \mid \mathit{\Pi} = t, B_j = i, C_j)\|p(A_j \mid B_j = i, C_j)) + \\ &\Pr[\mathit{\Pi} = t \mid A_j = i, C_j]\mathrm{D}(p(B_j \mid \mathit{\Pi} = t, A_j = i, C_j)\|p(B_j \mid A_j = i, C_j)).\end{aligned} \tag{3}$$

Our goal is to relate the expression $\sum_t \mathrm{E}[\mathrm{adv}(t, C_j)]$ to the advantage of the protocol, γ. For this purpose, we fix a transcript t where the output is one. We consider a fixed block j, and the matrix $N^t(C_j)$ that gives the probability of $\mathit{\Pi} = t$ for each pair of inputs for the parties $\mathcal{A}$ and $\mathcal{B}$, conditioned on the parts of their input C_j that we have already fixed (the probability here is taken over the randomness of the protocol and the remaining bits in the input of $\mathcal{A}$ and $\mathcal{B}$). To simplify notation we abbreviate $N^t(C_j)$ as N^t and write

$$N^t = \begin{bmatrix} N^t_{11}, N^t_{12}, N^t_{13} \\ N^t_{21}, N^t_{22}, N^t_{23} \\ N^t_{31}, N^t_{32}, N^t_{33} \end{bmatrix}.$$

Since $\mathit{\Pi}$ is a private-coin protocol, $\mathcal{A}$ and $\mathcal{B}$ can privately sample their remaining bits conditioned on C_j. Therefore (similarly to [2, Lemma 6.7]), N^t is a rank one matrix that can be expressed as $N^t = [a_1, a_2, a_3][b_1, b_2, b_3]^T$. In particular, b_i is the probability over $B_{1\dots j-1}$ and the private randomness of $\mathcal{B}$ that $B = B_{1\dots j-1}, B_j = i; B_{j+1\dots n}$ is in the rectangle for $\mathit{\Pi} = t$.

Relating the terms in (3) to N^t, observe that for $i = 1$, $\Pr[\mathit{\Pi} = t \mid B_j = 1, C_j] = N^t_{21} + N^t_{31} = a_2b_1 + a_3b_1$. Moreover, using the convention that $0/0 = 0$, $p(A_j \mid \mathit{\Pi} = t, B_j = 1, C_j)$ is a Bernoulli distribution with parameter $a_2b_1/(a_2b_1 + a_3b_1)$ (which we denote by $\mathfrak{B}_{a_2b_1/(a_2b_1+a_3b_1)}$), while $p(A_j \mid B_j = 1, C_j)$ is a Bernoulli distribution with parameter $1/2$ (as $A_j \in \{2,3\}$ is uniform). Consequently, we get the equality

$$\begin{aligned}\Pr[\mathit{\Pi} = t \mid B_j = 1, C_j]\mathrm{D}(p(A_j \mid \mathit{\Pi} = t, B_j = 1, C_j)\|p(A_j \mid B_j = 1, C_j)) = \\ (a_2b_1 + a_3b_1)\mathrm{D}(\mathfrak{B}_{a_2b_1/(a_2b_1+a_3b_1)}\|\mathfrak{B}_{1/2}).\end{aligned}$$

For any $x, y, z \in [0,1]$, define

$$IC(x, y, z) = (xy + xz)\mathrm{D}(\mathfrak{B}_{xy/(xy+xz)}\|\mathfrak{B}_{1/2}).$$

We generalize the above equality to all terms in (3), obtaining

$$\begin{aligned}\mathrm{adv}(t, C_j) = \frac{1}{3}(&IC(b_1, a_2, a_3) + IC(a_1, b_2, b_3) + IC(b_2, a_1, a_3) + \\ &IC(a_2, b_1, b_3) + IC(b_3, a_1, a_2) + IC(a_3, b_1, b_2)).\end{aligned} \tag{4}$$

Let P be the ordered set of triplets $(i_1, i_2, i_3) \in \{1, 2, 3\}^3$ such that i_1, i_2, i_3 are all distinct. Note that P contains 6 triples. Since $IC(x, y, z) = IC(x, z, y)$, we can write (4) as

$$\mathrm{adv}(t, C_j) = \frac{1}{6} \sum_{(i_1,i_2,i_3)\in P} (IC(a_{i_1}, b_{i_2}, b_{i_3}) + IC(b_{i_2}, a_{i_1}, a_{i_3})). \tag{5}$$

Each expression $IC(a_{i_1}, b_{i_2}, b_{i_3}) + IC(b_{i_2}, a_{i_1}, a_{i_3})$ can be thought of the information revealed by the protocol for a small AND gadget. The sum of the 6 expressions in $\mathrm{adv}(t, C_j)$ can be thought of as a "covering" of the matrix $N^t(C_j)$ with 6 AND gadgets.

For the following fact, we use the proof of [25, Lemma 4] to obtain a slightly better constant than the one obtained in [5]. Let $\phi = (1 + \sqrt{5})/2 \approx 1.618$ be the golden ratio. Recall that $\phi^2 = \phi + 1$.

Fact 2 *For any $x, y, z, u \in [0, 1]$, $IC(x, y, z)+IC(y, x, u) \geq \frac{1}{2\phi}(xz+yu-xy-zu)$.*

Proof. By Pinsker's inequality for Bernoulli distributions, we have

$$\mathrm{D}(\mathfrak{B}_{xy/(zy+xz)} \| \mathfrak{B}_{1/2}) \geq 2 \cdot \left(\frac{xy}{xy + xz} - \frac{1}{2}\right)^2 = \frac{1}{2}\left(\frac{z - y}{z + y}\right)^2$$

(if $xy + xz = 0$ the inequality follows from the definition $0/0 = 0$). Therefore,

$$\begin{aligned} IC(x, y, z) + IC(y, x, u) &= \\ (xy + xz)\mathrm{D}(\mathfrak{B}_{xy/(xy+xz)} \| \mathfrak{B}_{1/2}) + (yx + yu)\mathrm{D}(\mathfrak{B}_{yx/(yx+yu)} \| \mathfrak{B}_{1/2}) &\geq \\ \frac{1}{2}\left((xy + xz)\left(\frac{z - y}{z + y}\right)^2 + (yx + yu)\left(\frac{x - u}{x + u}\right)^2\right) &= \\ \frac{1}{2}\left(\frac{x}{y + z} \cdot (z - y)^2 + \frac{y}{x + u} \cdot (x - u)^2\right). \end{aligned}$$

Denote

$$\begin{aligned} R =& \frac{x}{y + z} \cdot (z - y)^2 + \frac{y}{x + u} \cdot (x - u)^2, \\ L =& xz + yu - xy - zu = (x - u)(z - y). \end{aligned}$$

Thus, in order to complete the proof we show that $R \geq L/\phi$. If L is not positive, then $R \geq L$ (since R is non-negative) and we are done. It remains to consider the case that $x \geq u$ and $z \geq y$ (the remaining case, $x \leq u$ and $z \leq y$, is symmetric). If $z \leq (2\phi+1)y$ (implying that $y/(y+z) \geq 1/(2\phi+2)$) then since $x/(x+u) \geq 1/2$, the product of the two terms of R is at least $(x - u)^2(z - y)^2/(4\phi + 4)$. Hence by the AM-GM inequality, $R \geq 2(x - u)(z - y)/\sqrt{4\phi + 4} = L/\sqrt{\phi + 1} = L/\phi$. If $z \geq (2\phi + 1)y$ then $z + y \leq (z - y)(\phi + 1)/\phi = \phi(z - y)$, hence the first term of R is at least $(x/\phi(z - y))(z - y)^2 = x(z - y)/\phi \geq L/\phi$. ■

We can now prove Theorem 6.

Proof (of Theorem 6). Combining (5) with Fact 2, we obtain

$$\mathrm{adv}(t, C_j) = \frac{1}{6} \sum_{(i_1,i_2,i_3)\in P} (IC(a_{i_1}, b_{i_2}, b_{i_3}) + IC(b_{i_2}, a_{i_1}, a_{i_3})) \geq$$

$$\frac{1}{12\phi} \sum_{(i_1,i_2,i_3)\in P} (a_{i_1} b_{i_3} + b_{i_2} a_{i_3} - a_{i_1} b_{i_2} - b_{i_3} a_{i_3}) =$$

$$\frac{1}{12\phi} \left(\sum_{i\neq i'} a_i b_{i'} - 2 \sum_{i\in\{1,2,3\}} a_i b_i \right) =$$

$$\frac{1}{12\phi} \left(\sum_{i\neq i'} (N^t_{ii'}(C_j) - 2 \sum_{i\in\{1,2,3\}} N^t_{ii}(C_j)) \right).$$

Therefore,

$$\sum_t \mathrm{E}[\mathrm{adv}(t, C_j)] \geq \frac{1}{12\phi} \sum_t \mathrm{E}\left[\sum_{i\neq i'} (N^t_{ii'}(C_j) - 2 \sum_{i\in\{1,2,3\}} N^t_{ii}(C_j)) \right] \geq \frac{1}{12\phi} \cdot 6\gamma = \gamma/(2\phi).$$

The second inequality follows since the advantage of Π is γ. In more detail, for some $\alpha \geq 0$, for each i, i' such that $i \neq i'$ we average the probability of outputting 1 over disjoint inputs and therefore we have $\sum_t \mathrm{E}[N^t_{ii'}(C_j)] \geq \alpha + \gamma$. On the other hand, for each $i \in \{1, 2, 3\}$ we average the probability of outputting 1 over inputs with intersection size of one and hence $\sum_t \mathrm{E}[N^t_{ii}(C_j)] \leq \alpha$.

Finally, combining with Fact 1 and (2),

$$\mathrm{H}[\Pi] \geq \frac{1}{2} \sum_j [\mathrm{I}(A_j; \Pi \mid C_j, B_j) + \mathrm{I}(B_j; \Pi \mid C_j, A_j)]] =$$

$$\frac{1}{2} \sum_j \sum_t \mathrm{E}[\mathrm{adv}(t, C_j)] \geq \frac{1}{4\phi} \sum_{j\in[N/3]} \gamma = \frac{1}{4\phi} \cdot \gamma N/3 \geq \gamma N/19.5,$$

concluding the proof. ■

A.4 The Proof of Theorem 3

We now use Theorem 6 to prove Theorem 3. The proof is based on the standard sparsification technique of [20].

Proof (of Theorem 3). We start with a public-coin protocol Π' for unique-disjointness with communication cost C' and advantage γ' and convert it into a private-coin protocol Π with communication cost at most $C = C' + 2.7 \log N + 17$ and advantage at least $\gamma \geq 0.99\gamma'$. By Theorem 6, we have $C \geq \gamma N/19.5$, or $C' + 2.7 \log N + 17 \geq \gamma' N/20$, implying that $C' \geq \gamma' N/20 - 2.7 \log N - 17 \geq \gamma' N/20 - 20 \log N$ and establishing Theorem 3 for $M = 20$.

Suppose Π' uses a string R as its randomness. For a parameter k, we pick k independent random strings $R_1, \ldots, R_k$, distributed as R. Fix an input (a, b)

such that $UDISJ(a, b) = 0$ and denote $\alpha = \mathrm{Err}_N^{\mathrm{UDISJ0}}(\Pi')$. Among $R_1, \ldots, R_k$, the expected number of strings R_i for which $\Pi'(a, b)$ errs with randomness R_i is at most αk. Hence, by a Chernoff bound, the probability that the number of strings for which $\Pi'(a, b)$ errs is more than $(\alpha + \gamma'/256)k = (1 + \gamma'/(256\alpha))\alpha k$ is at most $e^{-(\gamma'/(256\alpha))^2 \cdot \alpha k/3} > e^{-(\gamma')^2 \cdot k \cdot 2^{-16}}$. Since $\gamma' \geq 1/N$ (otherwise, the theorem is trivial), this probability is upper bounded by $e^{-k \cdot 2^{-16} N^{-2}}$. A similar bound can be shown for an input (a, b) such that $UDISJ(a, b) = 1$ by considering $\beta = \mathrm{Err}_N^{\mathrm{UDISJ1}}(\Pi')$.

We call a sequence of strings $R_1, \ldots, R_k$ *good* if for any (legal) input (a, b) to unique-disjointness, the fraction of strings for which Π' errs deviates from the corresponding error probability of Π' ($\mathrm{Err}_N^{\mathrm{UDISJ0}}(\Pi')$ or $\mathrm{Err}_N^{\mathrm{UDISJ1}}(\Pi')$) by at most $\gamma'/256$. Otherwise, the sequence is called *bad*. Taking a union bound over the (at most) 2^{2N} possible inputs, the probability that the sequence $R_1, \ldots, R_k$ is bad is at most $2^{2N} \cdot e^{-k \cdot 2^{-16} N^{-2}}$. Setting $k = 2^{17} N^{2.7}$ ensures that this probability is less than 1, and therefore there exists a good sequence of $k = 2^{17} N^{2.7}$ random strings, which we fix.

In the private-coin protocol Π, $\mathcal{A}$ first samples a uniform index $i \in [k]$ and sends it to $\mathcal{B}$ with its first message. This requires $\log k = 17 + 2.7 \log N$ additional bits of communication. The parties then run Π' with randomness R_i. Since $R_1, \ldots, R_k$ is good, then the advantage of Π is at least $\gamma' - \gamma'/256 - \gamma'/256 \geq 0.99\gamma'$, as claimed. ■