On the Communication Complexity of Key-Agreement Protocols

Iftach Haitner[1]
The Blavatnik school of computer science, Tel Aviv University, Israel
iftachh@cs.tau.ac.il

Noam Mazor[2]
The Blavatnik school of computer science, Tel Aviv University, Israel
joanrpublic@dummycollege.org

Rotem Oshman[3]
The Blavatnik school of computer science, Tel Aviv University, Israel
rotem.oshman@gmail.com

Omer Reingold[4]
Computer Science Department, Stanford University, USA
reingold@stanford.edu

Amir Yehudayoff[5]
Department of Mathematics, Technion-Israel Institute of Technology, Israel
amir.yehudayoff@gmail.com

Abstract

Key-agreement protocols whose security is proven in the *random oracle model* are an important alternative to protocols based on public-key cryptography. In the random oracle model, the parties and the eavesdropper have access to a shared random function (an "oracle"), but the parties are limited in the number of queries they can make to the oracle. The random oracle serves as an abstraction for black-box access to a symmetric cryptographic primitive, such as a collision resistant hash. Unfortunately, as shown by Impagliazzo and Rudich [STOC '89] and Barak and Mahmoody [Crypto '09], such protocols can only guarantee limited secrecy: the key of any ℓ-query protocol can be revealed by an $O(\ell^2)$-query adversary. This quadratic gap between the query complexity of the honest parties and the eavesdropper matches the gap obtained by the *Merkle's Puzzles* protocol of Merkle [CACM '78].

In this work we tackle a new aspect of key-agreement protocols in the random oracle model: their *communication complexity*. In Merkle's Puzzles, to obtain secrecy against an eavesdropper that makes roughly ℓ^2 queries, the honest parties need to exchange $\Omega(\ell)$ bits. We show that for protocols with certain natural properties, ones that Merkle's Puzzle has, such high communication is unavoidable. Specifically, this is the case if the honest parties' queries are uniformly random, or alternatively if the protocol uses non-adaptive queries and has only two rounds. Our proof for the first setting uses a novel reduction from the set-disjointness problem in two-party communication complexity. For the second setting we prove the lower bound directly, using information-theoretic arguments.

Understanding the communication complexity of protocols whose security is proven (in the random-oracle model) is an important question in the study of practical protocols. Our results and proof techniques are a first step in this direction.

[1] Supported by ERC starting grant 638121. Member of the Check Point Institute for Information Security.
[2] Supported by ERC starting grant 638121.
[3] Supported by the Israeli Centers of Research Excellence program 4/11 and BSF grant 2014256.
[4] Supported by NSF grant CCF-1749750.
[5] Supported by ISF grant 1162/15.

10th Innovations in Theoretical Computer Science (ITCS 2019).
Editor: Avrim Blum; Article No. 40; pp. 40:1–40:16

Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2012 ACM Subject Classification Theory of computation → Cryptographic protocols

Keywords and phrases key agreement, random oracle, communication complexity, Merkle's puzzles

Digital Object Identifier 10.4230/LIPIcs.ITCS.2019.40

Related Version A full version of the paper is available at [8], `https://eccc.weizmann.ac.il/report/2018/031/`.

Acknowledgements We thank Yuval Ishai for challenging us with this intriguing question, and Omer Rotem for very useful discussions.

1 Introduction

In a key-agreement protocol [5], two parties communicating over an insecure channel want to securely agree on a shared secret key, such that an eavesdropper observing their communication cannot find the key. For example, given a hash function $h : [n] \to [N]$ that is hard to invert, the players can execute the following protocol, called *Merkle's puzzles* [13]: we fix an arbitrary parameter $\ell \approx \sqrt{n}$, and the parties select uniformly random subsets $A = \{a_1, \ldots, a_\ell\}, B = \{b_1, \ldots, b_\ell\} \subseteq [n]$ (respectively) of size ℓ. We choose ℓ, n such that with constant probability there is a unique intersection, $|A \cap B| = 1$. The first party evaluates h on every element $a \in A$, and sends $h(a_1), \ldots, h(a_\ell)$ to the second party, which then looks for a unique element $b \in B$ such that $h(b) = h(a_i)$ for some $i \in [\ell]$. If found, the second party sends the index i to the first party and outputs b as the secret key; the second party outputs a_i as the secret key. Because h is a "good" hash function and $h(b) = h(a_i)$, it is likely that $b = a_i$, so the players output the same key. Moreover, since h is hard to invert, an eavesdropper that tries to find the secret key after seeing $h(a_1), \ldots, h(a_\ell), i$ must essentially compute h on the entire universe in order to invert h and find a_i. Thus, we have a quadratic gap between the work performed by the eavesdropper, which must compute $\Omega(\ell^2)$ hashes, and the work performed by the parties, which compute ℓ hashes each.

Ideally we would strive for an *exponential* gap between the work required to break the security of the protocol and the work of the honest parties. There are numerous candidate constructions of such key-agreement schemes, e.g., [17, 14, 1, 12], based on assumptions implying that *public-key encryption schemes* exist. A fundamental open question is whether we can design key-agreement protocols based on the security of symmetric primitives (e.g., collision resistant hash); the security of such primitives is believed to be more robust than public-key encryption. A very important step in this direction was made by Barak and Mahmoody[2] (following Impagliazzo and Rudich[10]): they showed that as long as the symmetric primitive is used as a black box, the quadratic gap achieved by Merkle's puzzles is the best possible.

The notion of "black box" is formalized by the *random oracle model*: instead of a concrete hash function h, we assume that the parties have access to a *random oracle* $\mathrm{F} : [n] \to [n]$, a perfectly random function. The random oracle is "the best hash function possible" (w.h.p.), so lower bounds proven in the random oracle model hold for any instantiation where the oracle is replaced by a one-way function. Thus, the lower bound of Barak and Mahmoody rules out any black-box key-agreement scheme from one-way functions that achieves a better than quadratic gap between the eavesdropper's work and the honest parties.

While a quadratic gap between the ℓ-query honest parties and the ℓ^2-query eavesdropper might not seem like much, and ideally we would wish for an exponential gap, on modern architecture it can yield a good enough advantage, assuming that security is preserved when the random oracle is replaced with a fixed hash function. For example, a consumer-level CPU (Intel Core i5-6600) can compute 5 million SHA-256 hashes per second, and specialized hardware for SHA-256 computation (for example, AntMiner S9) can compute 14×10^{12} hashes per second [3]. It follows that if the honest parties spend one second of computation on standard CPU, an attacker with specialized hardware can violate the security of Merkle's puzzles in less than a second. However, if the parties spend one second on specialized hardware, an attacker with specialized hardware has to spend more than $200,000$ years to break the scheme.

So, are Merkle's puzzles a practical and realistic key-agreement scheme? The answer is probably not: even setting aside the question of replacing the random oracle by a concrete hash function, in Merkle's puzzles, the honest parties *send each other* $\widetilde{\Omega}(\ell)$ *bits* to obtain security against an eavesdropper that makes roughly ℓ^2 queries. In our example above, if we instantiate Merkle's puzzles using SHA-256 for one second on specialized hardware, the first party would need to send more than 100 terabytes to the second party. A fundamental question is whether this high communication burden is inherent to secure key-agreement, and more generally, what is the communication cost of cryptographic protocols in the random oracle model and other oracle models. In this paper we initiate the study of the communication complexity of cryptographic protocols in the random-oracle model.

1.1 Our Results

We show that for random-oracle protocols with certain natural properties, the high communication incurred by Merkle's puzzles is unavoidable: in order to achieve security against an adversary that can ask $\Theta(\ell^2)$ queries, the two parties must exchange $\Omega(\ell)$ bits of communication. Specifically, we show that the bound above holds for protocols where the parties' queries are a uniformly random set, and also for two-round protocols that make non-adaptive (but arbitrary) queries.[6] We stress that a general lower bound for non-adaptive key-agreement protocols would imply a lower bound on the communication complexity of the set-intersection problem. This fact suggests that it is unlikely to find a simple proof for the general case.

To simplify the statements of our results, we focus here on key-agreement protocols whose *agreement* parameter, the probability that the players output the same key, is larger by some constant than their *secrecy* parameter, the probability that an eavesdropper can find the key.

Uniform-query protocols.

We say that a random-oracle protocol makes *uniform queries* if each party's oracle queries are a uniformly random set. We give the following lower bound on the communication complexity of such protocols.

▶ **Theorem 1** (lower bound on uniform-queries protocols, informal). *Any ℓ-uniform-query key-agreement protocol achieving non-trivial secrecy against $o(\ell^2)$-query adversaries has communication complexity $\Omega(\ell)$.*

This theorem is proved by a reduction from *set-disjointness*, a problem in communication complexity that is known to require high communication.

[6] These are both properties of Merkle's puzzles.

Two-round non-adaptive protocols.

An oracle protocol is said to make *non-adaptive* queries if the distribution of queries made by the players is fixed in advance, i.e., it is determined before the parties communicate with each other and does not depend on the oracle's answers. We give the following lower bound on the communication complexity of such protocols.

▶ **Theorem 2** (lower bound on two-message non-adaptive protocols, informal)**.** *Any two-message ℓ-query non-adaptive key-agreement protocol of non-trivial secrecy against q-query adversaries has communication complexity $\Omega(q/\ell)$.*

Once again this lower bound is nearly-tight with Merkle's puzzles, where $q = \Theta(\ell^2)$, and the communication cost is $\tilde{\Theta}(\ell)$.[7]

Following Barak and Mahmoody [2] and Impagliazzo and Rudich [10]), we prove this lower bound by presenting an eavesdropper that makes q queries and prevents the parties from exploiting the advantage they gain by their joint random oracle calls.

In [2], the communication cost of the protocol is not taken into account: their eavesdropper makes $O(\ell^2)$ queries and has high probability of finding *all* intersection queries (i.e., all queries that were asked by both players). In our case, if the protocol has communication cost C, then to prove Theorem 2, our eavesdropper must make only $O(C \cdot \ell)$ queries (to show the trade-off that $C = \Omega(q/\ell)$). If $C \ll \ell$, our eavesdropper makes much fewer queries than the eavesdropper in [2, 10], and in particular it cannot discover all the intersection queries. Instead, our eavesdropper asks only queries that the players *were able to learn* are in their intersection. If a query is in the intersection, but the players have not communicated this fact to each other, then the eavesdropper will not necessarily ask this query (unlike [2, 10]). Finding the correct definition for what it means to "learn" that a given query is in the intersection, and constructing an eavesdropper that makes only $O(C \cdot \ell)$ queries, are the main difficulty in our proof.

1.2 Related Work

Impagliazzo and Rudich [10] showed that the key of any ℓ-query key-agreement protocol in the random-oracle model can be revealed by an $\tilde{O}(\ell^6)$ query eavesdropper. Barak and Mahmoody [2] improve this bound and present an $O(\ell^2)$ query eavesdropper for this task, which shows that Merkle puzzles is optimal in this respect. Haitner, Omri, and Zarosim [9] used the machinery of [2] to relate the security of protocols that do not use a random oracle and solve tasks with no input, to the security of no-input protocols in the random-oracle model against an $O(\ell^2)$-query adversary. Finding limitations on the usefulness of random oracles for protocols that do take input seems to be a more difficult question. Chor and Kushilevitz [4] and Mahmoody et al. [11] made some progress in this direction. Finally, Haitner, Hoch, Reingold, and Segev [7] gave lower bounds on the communication complexity of statistically hiding commitments and single-server private information retrieval in a weaker oracle model that captures the hardness of one-way functions/permutation more closely than the random-oracle model.

[7] This theorem is also nearly-tight for any q, with a version of Merkle's puzzle, in which Alice is sending $\Theta(q/l)$ answers, from a universe of size $\Theta(q)$.

1.3 Organization

We begin by giving the formal definitions and notation used throughout the paper in Section 2. High-level overview of our proof techniques is given in Sections 3 and 4. For full proofs, see the full version of this paper in [8].

2 Preliminaries

2.1 Notations

We use calligraphic letters to denote sets, uppercase for random variables and lowercase for values. For $m \in \mathbb{N}$, let $[m] = \{1, \ldots, m\}$. For a random variable X, let $x \xleftarrow{R} X$ to denote that x is chosen according to X. Similarly, for a set S let $s \xleftarrow{R} \mathsf{S}$ to denote that s is chosen according to the uniform distribution over S. The support of the distribution D, denoted $\mathrm{Supp}(D)$, is defined as $\{u \in \mathcal{U} : \Pr_D[u] > 0\}$. The statistical distance between two distributions P and Q over a finite set $\mathcal{U}$, denoted $\mathrm{SD}(P, Q)$, is defined as $\frac{1}{2}\sum_{u\in\mathcal{U}} |\Pr_P[u] - \Pr_Q[u]|$, which is equal to $\max_{\mathcal{S}\subset\mathcal{U}}(\Pr_P[\mathcal{S}] - \Pr_Q[\mathcal{S}])$.

For a vector $\vec{X} = X_1, ..., X_n$ and an index $i \in [n]$, let $X_{<i}$ denote the vector $X_1, ..., X_{i-1}$ and $X_{\leq i}$ denote the vector $X_1, ..., X_i$. For a set of indexes $T = \{i_1, \ldots, i_k\} \subseteq [n]$ such that $i_1 < i_2 < \cdots < i_k$, let X_T denote the vector $X_{i_1}, \ldots, X_{i_k}$. Similarly, $X_{T,<i}$ denotes the vector $X_{T\cap\{1,\ldots,i-1\}}$. For a function f, let $f(\vec{X}) = (f(X_1), ..., f(X_n))$.

For random variables A and B we use $\mathrm{A}|_{\mathrm{B}=b}$ to denote the distribution of A condition on the event $\mathrm{B} = b$, and $\mathrm{A} \times \mathrm{B}$ to denote the product between the marginal distributions of A and B. When A is independent from B we write $\mathrm{A} \perp \mathrm{B}$ to emphasize that this is the case.

2.2 Interactive Protocols

A two-party protocol $\Pi = (\mathsf{A}, \mathsf{B})$ is a pair of probabilistic interactive Turing machines. The communication between the Turing machines A and B is carried out in rounds, where in each round one of the parties is active and the other party is idle. In the j-th round of the protocol, the currently active party P acts according to its partial view, writing some value on its output tape, and then sending a message to the other party (i.e., writing the message on the common tape). The communication transcript (henceforth, the transcript) of a given execution of the protocol $\Pi = (\mathsf{A}, \mathsf{B})$, is the list of messages m exchanged between the parties in an execution of the protocol, where $m_{1,\ldots,j}$ denotes the first j messages in m. A view of a party contains its input, its random tape and the messages exchanged by the parties during the execution. Specifically, A's view is a tuple $v_{\mathsf{A}} = (i_{\mathsf{A}}, r_{\mathsf{A}}, m)$, where i_{A} is A's input, r_{A} are A's random coins, and m is the transcript of the execution. Let $\mathrm{out}^{\mathsf{A}}$ denote the output of A in the end of the protocol, and $\mathrm{out}^{\mathsf{B}}$ B's output. Notice that given a protocol, the transcript and the outputs are deterministic function of the joint view $(i_{\mathsf{A}}, r_{\mathsf{A}}, i_{\mathsf{B}}, r_{\mathsf{B}})$. For a joint view v, let $\mathsf{trans}(v)$, $\mathrm{out}^{\mathsf{A}}(v)$ and $\mathrm{out}^{\mathsf{B}}(v)$ be the transcript of the protocol and the parties' outputs determined by v. For a distribution D we denote the distribution over the parties' joint view in a random execution of Π, with inputs drawn from D by $\Pi(D)$.

A protocol Π has r rounds, if for every possible random tapes for the parties, the number of rounds is exactly r. The Communication Complexity of a protocol Π, denoted as $\mathrm{CC}(\Pi)$ is the length of the transcript of the protocol in the worst case.

2.3 Oracle-Aided Protocols

An oracle-aided two-party protocol $\Pi = (\mathsf{A}, \mathsf{B})$ is a pair of interactive Turing machines, where each party has an additional tape called the oracle tape; the Turing machine can make a query to the oracle by writing a string q on its tape. It then receives a string ans (denoting the answer for this query) on the oracle tape. An oracle-aided protocol is ℓ-queries protocol if each party makes at most ℓ queries during each run of the protocol. In a *non-adaptive* oracle-aided protocol, the parties choose their queries before the protocol starts and before querying the oracle. A *uniform query* oracle-aided protocol, is a non-adaptive protocol in which the parties queries are chosen uniformly form a predetermined set.

2.4 Key-Agreement Protocols

Since we are giving lower bounds, we focus on single bit protocols.

▶ **Definition 3** (key-agreement protocol)**.** Let $0 \leq \gamma,\ \alpha \leq 1$ and $q \in N$. A two-party boolean output protocol $\Pi = (\mathsf{A}, \mathsf{B})$ is a (q, α, γ)-key-agreement relative to a function family $\mathcal{F}$, if the following hold:

Accuracy: Π has $(1-\alpha)$-accuracy. For every $f \in \mathcal{F}$:

$$\Pr_{v \xleftarrow{R} \Pi^f}\left[\mathrm{out}^{\mathsf{A}}(v) = \mathrm{out}^{\mathsf{B}}(v)\right] \geq 1 - \alpha.$$

Secrecy: Π has (q, γ)-secrecy. For every q-query oracle-aided algorithm E:

$$\Pr_{f \xleftarrow{R} \mathcal{F}, v \xleftarrow{R} \Pi^f}\left[\mathsf{E}^f(\mathsf{trans}(v)) = \mathrm{out}^{\mathsf{A}}(v)\right] \leq \gamma.$$

If $\mathcal{F}$ is a trivial function family (e.g., $\mathcal{F}$ contains only the identity function), then all correlation between the parties' view is implied by the transcript. Hence, an adversary that on a given transcript τ samples a random view for A that is consistent with τ, and outputs whatever A would upon this view, agrees with B with the same probability as does A. This simple argument yields the following fact:

For every $0 \leq \alpha \leq 1$ and $0 \leq \gamma < 1 - \alpha$, there exists no (q, α, γ)-key-agreement protocol relative to the trivial family.

2.5 Entropy and Information

In the proof we often need to measure differences between various distributions. For this purpose we use *f-divergences*: given a convex function $f : \mathbb{R} \to \mathbb{R}$ with $f(1) = 0$, and distributions P, Q, the *f-divergence of P from Q* is defined as

$$\mathsf{D}_f(\mathrm{P} \parallel \mathrm{Q}) = \sum_{q \in \mathrm{Q}} \Pr\left[\mathrm{Q} = q\right] f\left(\frac{\Pr\left[\mathrm{P} = q\right]}{\Pr\left[\mathrm{Q} = q\right]}\right).$$

Specifically, the two f-divergences we use in this paper are the *statistical distance*, obtained by taking $f(x) = |x-1|/2$, and the *KL divergence*, obtained by taking $f(x) = x \log x$. Each has its own nice properties and disadvantages: statistical distance is bounded in $[0, 1]$ but it is not additive, while KL divergence is additive but unbounded.

We frequently need to measure the "amount of dependence" between two random variables. Let $(\mathrm{X}, \mathrm{Y}) \sim P_{\mathrm{X},\mathrm{Y}}$ be random variables jointly distributed according to $P_{\mathrm{X},\mathrm{Y}}$, and let $P_{\mathrm{X}}, P_{\mathrm{Y}}$ be the marginal distribution of X and Y, respectively. Also, let $P_{\mathrm{X}} \times P_{\mathrm{Y}}$ be the product distribution where X and Y are sampled independently of each other, each from its marginal

distribution $P_{\mathrm{X}}, P_{\mathrm{Y}}$ (respectively). To quantify the dependence between X and Y, we measure the difference between their joint distribution and the product of the marginals: formally, we define

$$\mathrm{I}_f(\mathrm{X};\mathrm{Y}) = \mathsf{D}_f(P_{\mathrm{X},\mathrm{Y}} \parallel P_{\mathrm{X}} \times P_{\mathrm{Y}}).$$

This generalizes the usual notion of mutual information, which is the special case of I_f where we use KL divergence (i.e., when $f = x \log x$). For clarity, when we use KL divergence we omit the subscript f, and when using statistical distance, we use the notation I_{SD} (instead of $\mathrm{I}_{f(x)=|x-1|/2}$).

Finally, we also need the notion of *conditional mutual information*, which is simply the average mutual information between two variables X, Y, where the average is taken over a third random variable Z. Formally, let $(\mathrm{X}, \mathrm{Y}, \mathrm{Z}) \sim P_{\mathrm{X},\mathrm{Y},\mathrm{Z}}$. For any value z, let $P_{\mathrm{X},\mathrm{Y}|\mathrm{Z}=z}, P_{\mathrm{X}|\mathrm{Z}=z}, P_{\mathrm{Y}|\mathrm{Z}=z}$ be the joint distribution of X, Y and the marginals of X and Y, respectively, all conditioned on the event $\mathrm{Z} = z$.
Then we define $\mathrm{I}_f(\mathrm{X};\mathrm{Y}|\mathrm{Z}) = \mathrm{E}_{z\sim P_{\mathrm{Z}}}\left[\mathsf{D}_f(P_{\mathrm{X},\mathrm{Y}|\mathrm{Z}=z} \parallel P_{\mathrm{X}|\mathrm{Z}=z} \times P_{\mathrm{Y}|\mathrm{Z}=z})\right]$.

In the next sections we outline the strategy of the proofs.

3 Uniform-Query Protocols: Proof Outline

Our lower bound for uniform-query key-agreement protocols is proved via a reduction to *set disjointness*, a classical problem in two-party communication complexity.

In the set disjointness problem, we have two players, Alice and Bob. The players receive inputs $X, Y \subseteq [n]$, respectively, of size $|X| = |Y| = \ell$, and the players must determine whether $X \cap Y = \emptyset$. To do this, the players communicate with each other, and the question is how many bits they must exchange. It is known [16] that for any sufficiently large $n \in \mathbb{N}$, if the size of the sets is $\ell = n/4$, then the players must exchange $\Omega(n)$ bits to solve set disjointness, and this holds even for randomized protocols where the players have access to shared randomness and only need to succeed with probability 2/3. Here, we require high success probability *on any input*, not over some specific input distribution. We note that in the 2-party communication complexity model there is no random oracle.

The connection between set disjointness and key agreement comes from the fact that the only *correlation* between the parties' views in a key agreement protocol comes from the *intersection queries*, the queries that both players ask and Eve does not know. Indeed, if Alice asks $A \subseteq [n]$ and Bob asks $B \subseteq [n]$, and the random oracle is $F : [n] \to [n]$, then $F(A \setminus (A \cap B))$ and $F(B \setminus (A \cap B))$ are *independent of each other*. In particular, if $A \cap B = \emptyset$, then $F(A)$ and $F(B)$ are independent, and intuitively, in this case the players cannot securely agree on a secret key, because they have no advantage over the eavesdropper. On the other hand, if $A \cap B \neq \emptyset$, then the players can exploit the correlation induced by $F(A \cap B)$ to securely agree on a secret key. Thus, any secure key agreement protocol "behaves differently" depending on whether $A \cap B = \emptyset$ or not, and we can use this to solve the set disjointness problem.

Suppose that we are given a secure key-agreement protocol Π, where the players make ℓ uniformly-random queries to an oracle $\mathrm{F} : [n] \to [n]$. For simplicity we assume that the protocol has *perfect agreement*, that is, the players always output the same key, and that the security parameter is 3/4, that is, an eavesdropper has probability at most 3/4 of outputting the same key as the players. Our full proof does not make these assumptions.

Now, we want to construct from the key-agreement protocol Π, which uses a random oracle, a protocol Π' for set disjointness, *without* a random oracle (as usual in communication complexity). To this end, we consider two possible ways of simulating Π without an oracle:

- Λ_{Com}: the players use their shared randomness to simulate the oracle. They interpret the shared randomness as a random function $\mathrm{F} : [n] \to [n]$, and whenever Π wants to query some element $q \in [n]$, the players use $\mathrm{F}(q)$ as the oracle's answer.
- Λ_{Dist}: the players use their *private* randomness to simulate the oracle. Alice and Bob interpret their private randomness as random functions $\mathrm{F}_A, \mathrm{F}_B : [n] \to [n]$, respectively. Whenever Π indicates that Alice should query an element $q \in [n]$, she uses $\mathrm{F}_A(q)$ as the answer, while Bob uses $\mathrm{F}_B(q)$.

The first simulation, Λ_{Com}, is "perfect": it produces exactly the correct distribution of transcripts and outputs under our key-agreement protocol Π. In particular, the keys produced by the players in Λ_{Com} always agree, and an eavesdropper that sees the transcript of Λ_{Com} (but not the shared randomness) can find the key with probability at most $3/4$.

On the other hand, the second simulation Λ_{Dist} is "wrong", because the players do not use the same random function to simulate the random oracle. In fact, it is known that without shared randomness, secure key agreement is *impossible*, as an eavesdropper that sees the transcript can find the key with the same probability that the players have of agreeing with each other. Therefore there are two possible cases:

Agreement gap: The probability that the players agree on the key in Λ_{Dist} is at most $7/8$ (compared to one in Λ_{Com}), or

Secrecy gap: There is an eavesdropper E that guesses Alice's key in Λ_{Dist} with probability at least $7/8$ (compared to $3/4$ in Λ_{Com}).

(Instead of $7/8$ we could have used here any constant probability in $(3/4, 1)$, but in the full proof this choice depends on the agreement and security parameters of Π.)

We divide into cases, depending on which of the two gaps we have.

Agreement gap

Assume that the players agree with probability at most $7/8$ in Λ_{Dist}. For simplicity, let us make the stronger assumption that for any intersection size $c > 0$, the probability of agreement between the players is at most $7/8$, even *conditioned* on the event that $|A \cap B| = c$. A general key-agreement protocol might not satisfy this assumption, which complicates the full proof significantly; see the full version of this paper for the details.

So, we assumed that whenever the intersection is non-empty, the players agree with probability at most $7/8$. Observe, however, that when the intersection *is* empty ($A \cap B = \emptyset$), the distribution of transcript and outputs in Λ_{Dist} is the same as in Π: although each player uses a different random function, they never ask the same query, so there is no inconsistency. Therefore, conditioned on $A \cap B = \emptyset$, in Λ_{Dist} the players have perfect agreement (as in Π). In other words, Λ_{Dist} behaves very differently when $A \cap B = \emptyset$, in which case the players always agree on the key, compared to the general case, where the players agree with probability at most $7/8$. We use this fact to *check* whether $A \cap B = \emptyset$. Thus, by checking whether or not they got the same key in Λ_{Dist}, the players get an indication for whether or not $A \cap B = \emptyset$.

Our set disjointness protocol Π' is defined as follows. Given inputs $X, Y \subseteq [n]$, respectively, the players simulate Λ_{Dist} several times. In each simulation, the players agree on a random permutation $\sigma : [n] \to [n]$ using their shared randomness, and then the players simulate Λ_{Dist} using their permuted inputs as the query set; that is, Alice feeds $A = \sigma(X)$ to Λ_{Dist} as her query set, and Bob feeds $B = \sigma(Y)$ to Λ_{Dist} as his query set. Note that A, B are uniformly random, *subject to* having an intersection of size $|X \cap Y|$.

After each simulation of Λ_{Dist}, the players send each other the keys output under Λ_{Dist}, and check if they got the same key. Finally, they output "$X \cap Y = \emptyset$" iff they got the same key in all the simulations of Λ_{Dist}.

Since Λ_{Dist} has perfect agreement when there is no intersection, the players always succeed when $X \cap Y = \emptyset$. However, by assumption, whenever $X \cap Y \neq \emptyset$, the probability of agreement in Λ_{Dist} is at most 7/8, so if we repeat Λ_{Dist} sufficiently many times, the probability that all instances output the same key will be at most 1/3.

Secrecy gap

In this case we convert Λ_{Com} and Λ_{Dist} into a pair of protocols with an agreement gap, and then proceed as above.

Consider the protocol Λ'_{Dist} where the parties acts as in Λ_{Dist}, but at the end, Bob executes the eavesdropper E on the transcript, and outputs the key that E outputs. Define Λ'_{Com} analogously.

By assumption, E guesses Alice's output in Λ_{Dist} with probability at least 7/8, but in Λ_{Com} it succeeds with probability at most 3/4. Thus, in Λ'_{Dist} the players agree with probability at least 7/8, but in Λ'_{Com} they agree with probability at most 3/4; there is a gap of at least 1/8 between the probability of agreement in the two protocols (although they have switched roles and now Λ'_{Dist} has the higher agreement probability). Note also that Λ'_{Dist} does not have agreement probability 1, as we assumed for simplicity above, but our full proof can handle this case.

Formal statement

Here we give the formal statements that we prove for uniform-query protocols. Formally, our reduction is to set-disjointness over the distribution below, which known to be hard for low complexity protocols.

▶ **Definition 4** (hard distribution for set-disjointness). For $\ell \in \mathbb{N}$, let
$\mathcal{Q}^0_\ell = \{\mathcal{X}, \mathcal{Y} \subset [\ell] \colon |\mathcal{X}| = |\mathcal{Y}| = \lfloor \ell/4 \rfloor \ , \ \mathcal{X} \cap \mathcal{Y} = \emptyset\}$ and let
$\mathcal{Q}^1_\ell = \{\mathcal{X}, \mathcal{Y} \subset [\ell] \colon |\mathcal{X}| = |\mathcal{Y}| = \lfloor \ell/4 \rfloor , |\mathcal{X} \cap \mathcal{Y}| = 1\}$. Let D^0_ℓ and D^1_ℓ be the uniform distribution over $\mathcal{Q}^0_\ell$ and $\mathcal{Q}^1_\ell$ respectively, and let $D_\ell = \frac{3}{4} \cdot D^0_\ell + \frac{1}{4} \cdot D^1_\ell$.

Razborov [16] has shown that solving set-disjointness D_ℓ with small error require high communication complexity.

▶ **Theorem 5** (hardness of D_ℓ, [16]). *Exists $\epsilon > 0$ such that for every $\ell \in \mathbb{N}$ and a protocol Π that solves set-disjointness over D_ℓ with error ϵ, it holds that* $\mathrm{CC}(\Pi) \geq \Omega(\ell)$.

For a finite set $\mathcal{S}$, let $\mathcal{F}_{\mathcal{S}} = \left\{f : \mathcal{S} \mapsto \{0,1\}^*\right\}$ be the family of all functions from $\mathcal{S}$ to binary strings. Our reduction is stated in the following theorem.

▶ **Theorem 6** (from uniform-query key-agreement protocols to set-disjointness). *Assume exists an ℓ-uniform-query $(0, \alpha, \gamma)$-key agreement protocol relative to $\mathcal{F}_{\mathcal{S}}$, for some set $\mathcal{S}$, of communication complexity c. Then there exists a protocol for solving set-disjointness over D_ℓ with ϵ error and communication complexity $\frac{2^{15} \cdot \ell^4 \cdot \log 1/\epsilon}{|\mathcal{S}|^2 (1-\alpha-\gamma)^4} \cdot c$.*

Note that the above theorem holds also for protocols that are only secure against eavesdropper without access to the oracle. Combining theorems 5 and 6 yields the following bound on the communication complexity of uniform-query key-agreement protocols.

▶ **Theorem 7** (Main result for uniform-inputs protocols). *For any ℓ-uniform-query (q, α, γ)-key agreement protocol Π relative to $\mathcal{F}_{\mathcal{S}}$, it holds that* $\mathrm{CC}(\Pi) \in \Omega((1-\alpha-\gamma)^4 q^2/\ell^3)$.

What about general protocols?

It was important for our reduction to assume that the key-agreement protocol makes uniformly-random queries. Indeed, this reduction fails in the general case: consider the protocol where Alice and Bob always query 1, and output $\mathrm{F}(1)$ as their secret key. This protocol is completely insecure, since the eavesdropper can also query 1 and output $\mathrm{F}(1)$. But our reduction would not work for it, because the input distribution where both players get the set $\{1\}$ *is not hard for set disjointness* (indeed it is trivial). We see that the "hardness" of secure key-agreement is not necessarily that it is hard for the players to find their intersection queries, but that the eavesdropper should not be able to *predict* the intersection queries that the players use. Our second lower bound makes this intuition explicit and uses it to get a lower bound on two-round protocols with arbitrary (but non-adaptive) query distributions.

4 Two-Message Non-Adaptive Protocols: Proof Outline

In this section we describe a lower bound on the communication cost of any key-agreement protocol that makes non-adaptive queries and uses two rounds of communication: we show that any such protocol that makes ℓ queries and is secure against an adversary that makes q queries must send a total of $\Omega(q/\ell)$ bits. In particular, taking $q = \Theta(\ell^2)$, this shows that Merkle's puzzles is optimal in its communication cost. Formally, we show the following bound, where $\mathcal{F}_n$ is the family of all functions from $\{0,1\}^n$ to $\{0,1\}^n$:

▶ **Theorem 8** (Main theorem for two-message, non-adaptive protocols)**.** *For any $n \in \mathbb{N}$, the communication complexity of a two-message, non-adaptive, ℓ-query (q, α, γ)-key-agreement protocol relative to $\mathcal{F}_n$ is at least*

$$\frac{(1-\alpha-\gamma)^2 q}{50^2 \ell} - 6.$$

In this proof, we once again relate the parties' advantage over the eavesdropper to the information they gained about the intersection of their query sets. We show that to produce a shared key, the parties need to learn a lot of information about this intersection. Moreover, the query sets and their intersection need to be "unpredictable" (have high min-entropy) given the transcript, otherwise an eavesdropper could make the same queries and output the same key.

4.1 Some examples

Let us illustrate the ideas behind the lower bound by way of some examples.

Example 1

We already discussed the na ive example where both players query 1 and output $\mathrm{F}(1)$, and said that it is insecure because the eavesdropper can *predict* the intersection query. Here is another instantiation of this idea: Alice and Bob view the domain ℓ^2 as an $\ell \times \ell$ matrix, so that the oracle queries are represented by pairs $(i,j) \in [\ell]^2$. Alice chooses a row $a \in [\ell]$, and queries all the elements of the row (that is, all pairs (a,j) where $j \in [\ell]$); Bob chooses a column $b \in [\ell]$ and queries all the elements of the column (all pairs (i,b) where $i \in [\ell]$). Then, Alice sends a to Bob, who responds with $\mathrm{F}(a,b)$. From $\mathrm{F}(a,b)$, Alice can compute b, by finding the (w.h.p. unique) index j such that $\mathrm{F}(a,b) = \mathrm{F}(a,j)$. Both players output the first bit of b as the key.

This protocol is slightly less naïve than the previous one: now there are no queries that have high prior probability of being asked, and the index b of the query that determines the key is uniformly random a-priori. However, once Alice sends a to Bob, the game is up: Eve can also query row a and find b the same way Alice does.

We see that in addition to queries that have a high prior probability of being asked, Eve also needs to ask queries that have a high *posterior* probability of being asked, after she sees M_1. It turns out that this is enough: if we were to continue for more than 2 rounds, then Eve would also need to ask queries that become likely after seeing M_2, and so on, but to prove a 2-round lower bound, Eve does not need to ask these queries. Intuitively, if a query only becomes likely after M_2 is sent, then this is "too late" for it to be useful to the players, and Eve can ignore it.

Example 2

First, both players query 1. Then they carry out the protocol from Example 1, but all messages are "encrypted" by XOR-ing them with $\mathrm{F}(1)$.

From this example we see that Eve needs to be somewhat adaptive: when she decides what queries to ask after seeing M_1, she must incorporate the queries she asked before the first round (in this case, she would query 1). Essentially, when Eve tries to understand what the players have done in round i, she should take into account all the queries she made up to round i.

Should Eve be adaptive *inside* each round? In other words, after seeing M_1, should she ask all queries $\mathcal{E}_1$ that became likely, then compute which new queries are now likely given $\mathrm{M}_1, \mathcal{E}_1$, and so on, until she reaches a fixpoint?

It turns out that for our purposes here, because we consider non-adaptive protocols, Eve does not need to do this.

Heavy queries

Our attacker Eve tries to break the security of the protocol by asking all queries that are "somewhat likely" to be asked by the players; these queries are called *heavy queries*. Informally, a query $q \in \{0,1\}^n$ is *heavy* after round i if given the transcript up to round i (inclusive), and given Eve's queries up to round i, the probability that q is asked by one (or both) of the players exceeds some threshold δ which is fixed in advance.

More formally, the set $\mathcal{E}_i$ of heavy queries after round i is defined by induction on rounds, as follows: the a-priory heavy queries, $\mathcal{E}_0$, are given by

$$\mathcal{E}_0 = \{q \in \{0,1\}^n : \Pr[q \in X \cup Y] \geq \delta\}.$$

These are queries that are "somewhat likely" to be asked before the protocol begins. For $i > 0$, we define

$$\mathcal{E}_i = \mathcal{E}_{i-1} \cup \{q \in \{0,1\}^n : \Pr[q \in X \cup Y \mid M_{\leq i}, \mathrm{F}(\mathcal{E}_{i-1})] \geq \delta\}.$$

In other words, after round i, Eve asks all queries $q \in \{0,1\}^n$ that have probability at least δ of being queried by the players, given the messages $M_{\leq i}$ that Eve observed up to round i and the heavy queries she asked before, $\mathcal{E}_{i-1}$.

A simplified normal form for protocols

To simplify the proof of the lower bound, we first apply an easy transformation to the protocol: given a key agreement protocol Π, we construct a protocol Π', which has nearly the same communication and query complexity as Π, the same number of rounds, and the same agreement and nearly the same security parameters. But Π' also has the following properties: first, Π' has no a-priori heavy queries, that is, $\mathcal{E}_0 = \emptyset$; and second, the secret key output by Bob in Π' is the first bit of Bob's last query. This easy transformation is omitted in this overview.

Measuring the Players' Advantage Over Eve

As we saw in the examples above, the players' ability to produce a shared secret key is closely tied to how much information *the players* have that *Eve does not have* about the intersection of the query sets, $X \cap Y$.

To quantify this advantage, define the following random variables:

- $\mathrm{S}_i = (\mathrm{X} \cap \mathrm{Y}) \setminus \mathcal{E}_{i-1}$, the intersection queries that have not been asked by Eve.
- $\mathrm{F}(\mathrm{S}_i)$: the answers to the queries in S_i.
- $V_E^i = (\mathcal{E}'_i, \mathrm{F}(\mathcal{E}'_i))$: a subset of the heavy queries for the previous round, and the answers to them. Here, $\mathcal{E}'_i \subseteq \mathcal{E}_i$ is a subset that will be defined later (and depends on the round number i). For technical reasons, it is convenient to use only some of the heavy queries in some contexts; essentially, in some places in our proof, Eve uses only some of her power. This helps us avoid some unnecessary dependencies.

We measure the advantage gained by the players in round i by an expression of the form:

$$\mathrm{I}_f(\mathrm{S}_i, \mathrm{F}(\mathrm{S}_i); \mathrm{M}_i | \mathrm{Z}, \mathrm{V}_E^{i-1}, \mathrm{M}_{<i}), \tag{1}$$

where I_f is the information with respect to the f-divergence, $\mathrm{M}_i, \mathrm{M}_{<i}$ are the i-th message and the messages of rounds $1, \ldots, i-1$, Z is the query set of the player that sent M_i (either X or Y, depending on the round number i). Note that Eve uses her heavy queries from the previous rounds, V_E^{i-1}, to "try to understand" what is going on in the current round.

Intuitively, this expression measures how much information the i-th message conveys about the intersection queries and their answers, which Eve *cannot guess*. For this reason, the random variable S_i excludes intersection queries that were asked by Eve. Notice that on the right-hand side we condition on Eve's view (or on things Eve can sample): Eve has already seen the messages $\mathrm{M}_{<i}$ and asked the heavy queries $\mathrm{V}_E^{i-1} = (\mathcal{E}'_{i-1}, \mathrm{F}(\mathcal{E}'_{i-1}))$, and she can sample the queries Z, either X or Y, from the correct distribution given the transcript and her queries. Crucially, this does not require her to make any oracle queries: we do not require her to sample the answers F(Z), only the queries Z. In other words, Eve can *pretend* to be whichever player the query set Z belongs to, and by conditioning on her view, we essentially neutralize all the information that Eve can extract about the intersection. Thus, the expression in (1) measures the information the players gain about the intersection but that is hidden from Eve. [8]

Our proof consists of showing:

[8] As we said above, we use only some of Eve's heavy queries, $\mathcal{E}'_{i-1} \subseteq \mathcal{E}_i$, so this intuition is not completely accurate; specifically, when (1) is *large*, it does not mean that Eve cannot guess a lot about the intersection, because she could use the full set $\mathcal{E}_i$. However, when (1) is *small*, then indeed Eve knows almost as much about the intersection as the players do, because her view *includes* V_E^{i-1} (and possibly more).

Step I: After the first message M_1 is sent, the advantage gained is small, only $O(\delta|M_1|)$. For this part of the proof we use KL-divergence to measure the advantage.

Step II: After the second message M_2 is sent, the advantage is still small, only $O(\sqrt{\delta(|M_1|+|M_2|)})$. Here we use statistical distance to measure the advantage, for reasons we will explain below.

Step III: When the expression in (1) is small (i.e., the players only have a small "advantage"), then indeed, Eve can break the security of the protocol, by pretending to be one of the players and sampling the secret key that this player would output.

Next we explain in more detail how each step is carried out.

4.2 Outline of the Proof

Step III: How Eve breaks security

Let us start from the end: suppose that after the second round, the "advantage" is small:

$$I_f(S_2, F(S_2); M_2 | Y, M_1, V_E^1) \leq \beta,$$

where $\beta = O(\sqrt{\delta(|M_1|+|M_2|)}) \ll 1$. Here, the advantage is measured in statistical distance (that is, we take $f(t) = |t-1|/2$). We want to show that Eve can break the security of the protocol, by guessing the secret key.

As we said, Eve's strategy is to "pretend" that she is Bob, and sample Bob's output, out^B.

In a general protocol, to do this, Eve needs to sample Bob's queries Y and the answers F(Y), and then she can compute $out^B = out^B(Y, F(Y), M_1, M_2)$. However, recall that we transformed the protocol so that out^B is a fixed function of Y; therefore, Eve in fact needs to do nothing clever, only sample Y given her view $M_1, M_2, \mathcal{E}_1, F(\mathcal{E}_1)$ and compute out^B from Y.

We need to show that Eve's key is close to the correct distribution, the one used by the players. In general, if too much communication is allowed, this is not true, as shown by the following example.

▶ **Example 9.** In Merkle's puzzles, Alice's message is F(X), and Bob responds with F(s), where $s \in X \cap Y$ is some intersection query. The original secret key (before our transformation) is the first bit s^1. After our transformation, the secret key is $Y^1_{\ell+1}$, and as part of M_2, Bob sends Alice the bit $b = s^1 \oplus Y^1_{\ell+1}$ so that she can extract $Y^1_{\ell+1}$.

From Alice's perspective, given X, F(X), Bob's message $M_2 = F(s), b$ fixes $Y^1_{\ell+1}$ to the value $b \oplus s^1$. (We ignore here the tiny probability that s cannot be uniquely computed from X, F(X) and F(s), i.e., the probability of a collision in F.) However, from Eve's perspective, because she does not know X, F(X) and she asks no queries (there are no heavy queries in Merkle's puzzles), the intersection element s remains uniformly random. When Eve samples $Y^1_{\ell+1}$ given M_1, M_2 and her non-existent heavy queries, the result is random, and completely independent from the true secret key.

We need to show that when the players' advantage is small, then the example above cannot happen, and Eve's key agrees with the players' w.h.p. To this end, we are interested in the difference between Eve's "pretend distribution", and the true distribution that the players use to produce the key: if the two distributions are close, then Eve's chances of guessing the right secret key are roughly the same as Bob's. The *only difference* between these two distributions is that given M_1, M_2 and $\mathcal{E}_1, F(\mathcal{E}_1)$ (which the players do not use),

- The players' keys are produced according to the *joint distribution* $(\text{out}^{\mathsf{A}}, \text{out}^{\mathsf{B}})$, and in particular, both players have the same answers $\text{F}(\text{S}_2)$ to the non-heavy intersection queries $\text{S}_2 = (\text{X} \cap \text{Y}) \setminus \mathcal{E}_1$.
- Eve's pretense that she is Bob is carried out *independently* from Alice's view: Eve cannot use the true intersection queries (which she does not know), only what she has learned about them from $\text{M}_1, \text{M}_2, \text{V}_E^i$. The joint distribution of Alice and Eve's keys is therefore given by the product distribution $\text{out}^{\mathsf{A}} \times \text{out}^{\mathsf{B}}$.

So, we would like to bound the difference between the joint distribution and the product distribution, i.e.,

$$\text{I}_f(\text{out}^{\mathsf{A}}; \text{out}^{\mathsf{B}} | \text{M}_1, \text{M}_2, \mathcal{E}_1, \text{F}(\mathcal{E}_1)).$$

Given the conditioning, Alice's output out^{A} is a function of her view, $\text{X}, \text{F}(\text{X})$. Also, we assumed that Bob's output is a function of his queries Y. Therefore,

$$\text{I}_f(\text{out}^{\mathsf{A}}; \text{out}^{\mathsf{B}} | \text{M}_1, \text{M}_2, \mathcal{E}_1, \text{F}(\mathcal{E}_1)) \leq \text{I}_f(\text{X}, \text{F}(\text{X}); \text{Y} | \text{M}_1, \text{M}_2, \mathcal{E}_1, \text{F}(\mathcal{E}_1)). \tag{2}$$

Now we need to show that given Eve's view, the dependence between $\text{X}, \text{F}(\text{X})$ and Y is bounded in terms of the advantage:

$$\text{I}_f(\text{X}, \text{F}(\text{X}); \text{Y} | \text{M}_1, \text{M}_2, \mathcal{E}_1, \text{F}(\mathcal{E}_1)) \leq \text{I}_f(\text{S}_2, \text{F}(\text{S}_2); \text{M}_2 | \text{M}_1, \text{Y}, \mathcal{E}_1 \cap \text{Y}, \text{F}(\mathcal{E}_1 \cap \text{Y})). \tag{3}$$

This proof is somewhat tedious; it relies on the fact that M_2 is a function of M_1, Y and $\text{F}(\text{Y})$, and on the fact that $\text{X}, \text{F}(\text{X})$ are independent of $\text{Y}, \text{F}(\text{Y})$ given the intersection queries and answers, $\text{S}_2, \text{F}(\text{S}_2)$ and $\mathcal{E}_1, \text{F}(\mathcal{E}_1)$. Intuitively, all the dependence between $\text{X}, \text{F}(\text{X})$ and Y "flows through" what the players learn about the intersection, and the proof of (3) formalizes this intuition.

Step I: Bounding the advantage after the first round

For the first round, we analyze the players' advantage in terms of KL-divergence, and bound

$$\text{I}(\text{S}_1, \text{F}(\text{S}_1); \text{M}_1 | \text{X}).$$

Notice that we do not use Eve at this point, because we eliminated any a-priory heavy queries, so there is nothing Eve needs to query in order to "understand" M_1. For the same reason, $\text{S}_1 = \text{X} \cap \text{Y}$ (there are no heavy queries to remove from the intersection).

We claim that

$$\text{I}(\text{S}_1, \text{F}(\text{S}_1); \text{M}_1 | \text{X}) \leq \delta |\text{M}_1|. \tag{4}$$

This is not hard to see: suppose $\text{X} = x$. Because we got rid of the a-priori heavy queries, every individual query $q \in x$ has probability at most δ of being asked by Bob (otherwise, q would be heavy). Therefore, for every $q \in x$, we have $\Pr[q \in \text{S}_1 | \text{X} = x] \leq \delta$. Because M_1 is generated by Alice without knowing S_1, and every query is in S_1 only w.p. at most δ, intuitively, the information in M_1 "only applies" to the queries in S_1 with probability δ. Therefore the information that M_1 gives about $\text{S}, \text{F}(\text{S}_1)$ is at most $\delta |\text{M}_1|$.

The actual proof involves a Shearer-like argument for mutual information, similar to the ones used in [6, 15].

Step II: Bounding the advantage after the second round

Now we must bound the advantage the players gain after the second round, and show that

$$\mathrm{I}_{SD}(\mathrm{S}_2, \mathrm{F}(\mathrm{S}_2); \mathrm{M}_2 | \mathrm{Y}, \mathrm{M}_1, \mathcal{E}_1 \cap \mathrm{Y}, \mathrm{F}(\mathcal{E}_1 \cap \mathrm{Y})) = O(\sqrt{|\mathrm{M}_1| + |\mathrm{M}_2|}). \tag{5}$$

As we said, we switch here to using statistical distance, and we will see why below.

Following the first round, we know that not much is known about the intersection, because Alice's message M_1 did not convey a lot of information about it. So, our proof here proceeds in two steps: first, we "pretend" that *nothing* is known about the intersection, and consider the distribution μ' where given M_1 the distribution of $\mathrm{Y}, \mathrm{F}(\mathrm{Y})$ is completely independent from X. We show that under μ', Bob's message M_2 would only convey $\delta|\mathrm{M}_2|$ bits of information about the intersection. This is very similar to the analysis of the first round, and it is also carried out using KL-divergence. Formally, we show that for the distribution μ' where $\mathrm{Y}, \mathrm{F}(\mathrm{Y})$ are drawn independently of of X, we have

$$\mathrm{I}^{\mu'}(\mathrm{S}_2, \mathrm{F}(\mathrm{S}_2); \mathrm{M}_2 | \mathrm{Y}, \mathrm{M}_1, \mathcal{E}_1 \cap \mathrm{Y}, \mathrm{F}(\mathcal{E}_1 \cap \mathrm{Y})) \leq \delta|\mathrm{M}_2|. \tag{6}$$

The proof relies on the fact that we excluded heavy queries from S_2 (recall that $\mathrm{S}_2 = (\mathrm{X} \cap \mathrm{Y}) \setminus \mathcal{E}_1$), so given the conditioning, any query in Y can only belong to S_2 with probability at most δ.

However, μ' is not the real distribution: given M_1, we do know a little about the intersection, so $\mathrm{Y}, \mathrm{F}(\mathrm{Y})$ are not completely independent from X. Our next step is to switch to statistical distance, and show that the real distribution μ (where X, Y are not independent) and μ' (where they are) are close to each other. Therefore, what we showed for μ' is also true for μ, with the addition of a small penalty corresponding to the distance between μ and μ'.

Formally, we prove that

$$\begin{aligned} &\mathrm{I}^{\mu}_{SD}(\mathrm{S}_2, \mathrm{F}(\mathrm{S}_2); \mathrm{M}_2 | \mathrm{Y}, \mathrm{M}_1, \mathcal{E}_1 \cap \mathrm{Y}, \mathrm{F}(\mathcal{E}_1 \cap \mathrm{Y})) \\ &\leq O\left(\mathrm{I}^{\mu'}_{SD}(\mathrm{S}_2, \mathrm{F}(\mathrm{S}_2); \mathrm{M}_2 | \mathrm{Y}, \mathrm{M}_1, \mathcal{E}_1 \cap \mathrm{Y}, \mathrm{F}(\mathcal{E}_1 \cap \mathrm{Y})) + \mathsf{D}_{SD}(\mu' \parallel \mu)\right) \end{aligned} \tag{7}$$

Under μ', by (6) and Pinsker's inequality, we have:

$$\mathrm{I}^{\mu'}_{SD}(\mathrm{S}_2, \mathrm{F}(\mathrm{S}_2); \mathrm{M}_2 | \mathrm{Y}, \mathrm{M}_1, \mathcal{E}_1 \cap \mathrm{Y}, \mathrm{F}(\mathcal{E}_1 \cap \mathrm{Y})) \leq \sqrt{\delta|\mathrm{M}_2|}. \tag{8}$$

So, under μ' the expected amount of information revealed is small.

Next, we bound the difference between μ and μ'. We show that:

$$\mathrm{I}(\mathrm{Y}, \mathrm{F}(\mathrm{Y}); \mathrm{X} | \mathrm{M}_1) \leq \mathrm{I}(\mathrm{S}_1, \mathrm{F}(\mathrm{S}_1); \mathrm{M}_1 | \mathrm{X}).$$

This is quite similar to the proof of Step III above – here we do use standard mutual information, so the proof uses the chain rule, just as we did above. Since we have shown in Step I that $\mathrm{I}(\mathrm{S}_1, \mathrm{F}(\mathrm{S}_1); \mathrm{M}_1 | \mathrm{X}) \leq \delta|\mathrm{M}_1|$, we conclude using Pinsker's inequality that

$$\mathsf{D}_{SD}(\mu' \parallel \mu) \leq \sqrt{\mathsf{D}_{KL}(\mu' \parallel \mu)} \leq \sqrt{\delta|\mathrm{M}_1|}. \tag{9}$$

Together, (8) and (9) are the ingredients we need to apply (7), and obtain:

$$\begin{aligned} &\mathrm{I}^{\mu}_{SD}(\mathrm{S}_2, \mathrm{F}(\mathrm{S}_2); \mathrm{M}_2 | \mathrm{Y}, \mathrm{M}_1, \mathcal{E}_1 \cap \mathrm{Y}, \mathrm{F}(\mathcal{E}_1 \cap \mathrm{Y})) \\ &\leq O\left(\mathrm{I}^{\mu'}_{SD}(\mathrm{S}_2, \mathrm{F}(\mathrm{S}_2); \mathrm{M}_2 | \mathrm{Y}, \mathrm{M}_1, \mathcal{E}_1 \cap \mathrm{Y}, \mathrm{F}(\mathcal{E}_1 \cap \mathrm{Y})) + \mathsf{D}_{SD}(\mu' \parallel \mu)\right) \\ &\leq O(\sqrt{\delta(|\mathrm{M}_1| + |\mathrm{M}_2|)}). \end{aligned}$$

References

1. Miklós Ajtai and Cynthia Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 284–293. ACM, 1997.
2. B. Barak and M. Mahmoody. Merkle Puzzles Are Optimal - An $O(n^2)$-Query Attack on Any Key Exchange from a Random Oracle. In *Advances in Cryptology – CRYPTO '09*, pages 374–390, 2009.
3. Daniel J Bernstein and Tanja Lange. eBACS: ECRYPT benchmarking of cryptographic systems. https://bench.cr.yp.to, accessed 15 May 2018. URL: `https://bench.cr.yp.to`.
4. Benny Chor and Eyal Kushilevitz. A zero-one law for boolean privacy. *SIAM Journal on Discrete Mathematics*, 4(1):36–47, 1991.
5. Whitfield Diffie and Martin E. Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.
6. Anat Ganor, Gillat Kol, and Ran Raz. Exponential separation of information and communication for boolean functions. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 557–566. ACM, 2015.
7. Iftach Haitner, Jonathan J. Hoch, Omer Reingold, and Gil Segev. Finding Collisions in Interactive Protocols - Tight Lower Bounds on the Round and Communication Complexities of Statistically Hiding Commitments. *SIAM Journal on Computing*, 44(1):193–242, 2015. Preliminary version in *STOC'07*.
8. Iftach Haitner, Noam Mazor, Rotem Oshman, Omer Reingold, and Amir Yehudayoff. On the Communication Complexity of Key-Agreement Protocols. In *Electronic Colloquium on Computational Complexity (ECCC)*, volume 25, page 31, 2018.
9. Iftach Haitner, Eran Omri, and Hila Zarosim. Limits on the usefulness of random oracles. *Journal of Cryptology*, 29(2):283–335, 2016.
10. Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing (STOC)*, pages 44–61. ACM Press, 1989.
11. Mohammad Mahmoody, Hemanta K Maji, and Manoj Prabhakaran. Limits of random oracles in secure computation. *arXiv preprint*, 2012. `arXiv:1205.3554`.
12. Robert J McEliece. A public-key cryptosystem based on algebraic. *Coding Thv*, 4244:114–116, 1978.
13. Ralph C. Merkle. Secure Communications over Insecure Channels. In *SIMMONS: Secure Communications and Asymmetric Cryptosystems*, 1982.
14. Michael O Rabin. Digitalized signatures and public-key functions as intractable as factorization. Technical report, Massachusetts Inst of Tech Cambridge Lab for Computer Science, 1979.
15. Anup Rao and Makrand Sinha. Simplified Separation of Information and Communication. In *Electronic Colloquium on Computational Complexity (ECCC)*, volume 22 (57), pages 2–3, 2015.
16. Alexander A. Razborov. On the distributional complexity of disjointness. *Theoretical Computer Science*, 106(2):385–390, 1992.
17. Ronald L. Rivest, Adi Shamir, and Leonard M. Adelman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.