On the Privacy of Sublinear-Communication Jaccard Index Estimation via Min-hash Sketching

Seung Geol Choi, Dana Dachman-Soled, Mingyu Liang, Linsheng Liu, and Arkady Yerukhimovich

1 United States Naval Academy
2 University of Maryland, College Park
3 George Washington University

Abstract. The min-hash sketch is a well-known technique for low-communication approximation of the Jaccard index between two input sets. Moreover, there is a folklore belief that min-hash sketch based protocols protect the privacy of the inputs. In this paper, we investigate this folklore to quantify the privacy of the min-hash sketch.

We begin our investigation by considering the privacy of min-hash in a centralized setting where the hash functions are chosen by the min-hash functionality and are unknown to the participants. We show that in this case the min-hash output satisfies the standard definition of differential privacy (DP) without any additional noise. This immediately yields a privacy-preserving sublinear-communication semi-honest 2-PC protocol based on FHE where the hash function is evaluated homomorphically.

To improve the efficiency of this protocol, we next consider an implementation in the random oracle model. Here, the protocol participants jointly sample public prefixes for domain separation of the random oracle, and locally evaluate the resulting hash functions on their input sets. Unfortunately, we show that in this public hash function setting, the min-hash output is no longer DP. We therefore consider the notion of distributional differential privacy (DDP) introduced by Bassily et al. (FOCS 2013). We show that if the honest party’s set has sufficiently high min-entropy then the output of the min-hash functionality achieves DDP, again without any added noise. This yields a more efficient semi-honest two-party protocol in the random oracle model, where parties first locally hash their input sets and then perform a 2PC for comparison.

By proving that our protocols satisfy DP and DDP respectively, our results formally confirm and qualify the folklore belief that min-hash based protocols protect the privacy of their inputs.

1 Introduction

Min-hash sketch. The min-hash sketch is a simple and well-known technique to produce an unbiased estimate of the Jaccard index [8,32]. The Jaccard index [29] is a similarity measure between two sets $A$ and $B$, denoted $J(A, B)$, which is defined as the fraction of the elements in the intersection of $A$ and $B$ divided by the number of elements in the union of them. That is, $J(A, B) = \frac{|A \cap B|}{|A \cup B|}$.

The Jaccard index has seen wide application for clustering of websites and...
documents\cite{59}, community identification \cite{43}, DNA matching \cite{13}, and machine learning \cite{45,30}.

However, computing the Jaccard index exactly, especially when the input sets are large, can be costly. The min-hash sketch allows communication-efficient approximation. The basic idea behind the min-hash sketch is to apply a random permutation $\pi$ to both sets $A$ and $B$ and then to see whether the last item in both sets (under this permutation) is the same. Since this permutation is applied over all elements in $A \cup B$, it is easy to see that the last item will be the same exactly when the last item over $A \cup B$ is in both $A$ and $B$. Specifically,

$$\Pr[\text{last item in } \pi(A) = \text{last item in } \pi(B)] = \frac{|A \cap B|}{|A \cup B|} = J(A, B)$$

Thus, to get an unbiased approximation of the Jaccard index, it suffices to repeat this procedure with sufficiently many random permutations.

**Motivation.** Due to its simplicity and efficiency, the min-hash sketch has become a very popular tool to approximate the Jaccard index. Moreover, since the min-hash sketch only needs to compare the last items in the permuted sets, it has been a key building block when maintaining privacy of the input sets is important, e.g., if the input sets represent fingerprints, DNA, or medical records. There are two classes of solutions for privacy-preserving min-hash. The first class of solutions (e.g. \cite{13,37,22}) considers how to compute the min-hash in a two-party setting, where the parties do not trust each other with their private inputs. The goal of these works is to design secure two-party computation protocols for computing the min-hash sketch as efficiently as possible, but they generally do not consider the implication of revealing the final result. The second line of work (e.g. \cite{46,47,1}) considers how to make the min-hash itself more privacy-preserving. Specifically, these works consider adding noise to the output of the min-hash algorithm to guarantee differential privacy (DP) of individual items in the input sets.

These works serve as the starting point for our paper. We aim to design protocols for two-party secure estimation of the Jaccard index that additionally preserve the privacy of individual items inside the input sets. But, unlike prior work, we make a very important restriction. To minimize the error of the approximation and to guarantee reproducibility of the results, we do not allow our protocol to add any additional noise to the output of the min-hash protocol. This goal forces us to ask the following critical question:

*Does the min-hash algorithm itself provide privacy guarantees for its inputs?*

**Differential privacy of the min-hash sketch.** Somewhat surprisingly, we show that the answer to this question is yes. Specifically, we show that under certain application scenarios and restrictions on the input sets, the error of the min-hash approximation of the Jaccard index is actually sufficient to achieve differential privacy. Essentially, the error of the sketch acts as noise to protect the
privacy of the inputs. Similar observations that sketching algorithms inherently preserve privacy have previously been shown for the Johnson-Lindenstrauss sketch [7], the LogLog sketch [12, 42], and other sketches [44].

To get an understanding of the underlying privacy of the min-hash protocol, we first consider a setting with a set of private permutations chosen by the functionality (or a trusted curator in the standard differential privacy setting) and unknown to the parties. The functionality then uses these permutations to find the minimum item in the two sets and output the total number of times the minimums match. Standard differential privacy in this setting requires that conditioned on knowledge of $A$ and all but one element of $B$ (denoted by $x^*$), the probability that the functionality outputs any value out when $x^* \in B$ versus when $x^* \notin B$ differs by a factor of at most $e^\epsilon$ with all but negligible probability.

We note that min-hash is not differentially private in this setting if $A \cap B$ is either too large or too small. For example, if $|A \cap B| = 0$ when $x^* \notin B$ and 1 when $x^* \in B$, then min-hash always outputs 0 in the first case and outputs a count $\geq 1$ with noticeable probability in the second. We prove the following theorem showing that when this is not the case the min-hash output is differentially private:

**Theorem 1 (Informal).** If the size of the intersection is a constant fraction of the size of $A$ and $B$, then the output of this min-hash protocol is $(\epsilon, \delta)$-DP for negligible $\delta$.

We stress that this theorem crucially relies on the fact that the parties, and the adversary, do not have any information about the chosen permutations, and cannot learn the evaluation of the permutations on their own inputs. This, of course, poses a particular challenge if one wants to deploy this protocol without relying on a trusted curator or functionality. However, despite this challenge, we note that using fully-homomorphic encryption [24] to evaluate the permutations and compare the minimums, it is possible to build a protocol with sublinear communication. Since it securely realizes the functionality (with semi-honest security), the protocol would be computational DP [31, 36, 39]. Additionally, since the error is unbiased and from an easily sampleable distribution, this protocol is also a secure approximation [23] of the Jaccard index in that it leaks no information about the input sets beyond the Jaccard index.

**Distributional DP in the two-party setting.** However, the need to keep the permutations private in Theorem 1 forces our initial protocol to evaluate the permutations inside of FHE for each input, resulting in an inefficient protocol in practice.

Therefore, we consider what happens if we do not keep the permutations private from the parties. The major benefit of revealing the permutations is that the parties can then apply them to their own sets locally, and only perform secure computation to compare the minimum elements and count the number of matches. Thus, allowing for much simpler protocols with sublinear communication and only $O(k)$ secure gate evaluations where $k$ is the number of permutations which is far smaller than the input size.
Unfortunately, there is a problem with the above approach. It is easy to see that, if we reveal the permutations, the min-hash protocol is not in fact differentially private, as defined above. The problem stems from the fact that in the standard DP setting, we assume that the adversary knows all of the inputs (in this case, all entries in both sets $A$ and $B$) except for some input $x^*$ and wants to determine, from the output of the computation, whether $x^*$ was in the other party's set. If the permutations are known, then the adversary can reconstruct the min-hash exactly both for the case when $x^*$ is in the set and when it is not, and then see which of these matches the output it received. Since the min-hash protocol provides a good approximation of the Jaccard index, the adversary will be able to exactly determine whether or not $x^* \in B$ with noticeable probability.

The above gives a counter-example against the differential privacy of min-hash in the public permutation setting. However, does it really give rise to a realistic privacy breach?

The above attack works only if the adversary knows the entirety of both sets $A, B$ and just tries to distinguish whether $x^* \in B$ or not. Realistically, the adversary would not know the entire input of the honest party. Indeed, parties would only want to perform a secure computation on their private inputs in the first place if they did not already know what the answer would be. In the context of min-hash, this means that the adversary does not already know a good portion of the honest party's input set. More precisely, this motivates the assumption that given the adversary's set (and even the intersection between the two sets), the honest party's set still has sufficiently high min-entropy. With this assumption, we turn to the tool of distributional DP (DDP) \cite{DworkMNTT14} which allows us to analyze differential privacy when the distribution of inputs has sufficient uncertainty.

We begin with a relatively strong assumption on the amount of uncertainty the adversary has about the honest set. Specifically, we assume that every element that is not in the intersection is highly unpredictable (i.e., has a high amount of min-entropy), even conditioned on all the other set elements. Under this assumption, we prove the following theorem:

**Theorem 2 (Informal).** If each non-intersecting item has sufficiently high min-entropy, revealing the hash functions\footnote{We use cryptographic hash functions to instantiate the permutations in the random oracle model.} together with the min-hash counts preserves $(\epsilon, \delta)$-DDP for negligible $\delta$, as long as the size of the intersection is a constant fraction of the size of $A$ and $B$.

Not surprisingly, the proof of this Theorem (given in Section 4.2) leverages the fact that when each element has individual high min-entropy, hashing each element acts as a strong randomness extractor, thus resulting in sufficient random noise for privacy.

**DDP over a polynomial-size universe.** However, this assumption that every item has high min-entropy is quite strong. For example, consider the setting where each item in $B$ is chosen from a polynomial-size universe. In this case,
while individual items cannot have much min-entropy, the honest party’s set may still collectively have high min-entropy as long as it is large enough. Thus, for our third result, we analyze what happens under this weaker assumption that only the full honest set, instead of each individual item, has high min-entropy.

Note that in this case, we cannot apply the hash function as randomness extractor technique. This is because in order to guarantee that the randomness extractor yields output that is negligibly close to uniform, we must lose super-logarithmic in $n$ bits of entropy from each input. However, in the case we are currently considering, each element has at most $O(\log n)$ bits of min-entropy.

Further, we in fact have no guarantee that each element has individually high min-entropy (since the elements are not necessarily independent), but only that the total min-entropy of the non-intersection items is high. Nevertheless, we show the min-hash protocol still achieves DDP, by proving a new strong chain rule for min-entropy (see Section 7).

Specifically, we consider the following class of distributions $C$ over secret sets $R$ of size $n$:

- Let $\mathcal{U}$ be a universe of polynomial size $n \cdot \ell$, where $\ell = \Omega(n^3)$.
- $R$ is chosen uniformly from all subsets of $\mathcal{U}$ of size $n$.
- Arbitrary leakage $L = L(R)$ is computed on $R$, we require that the length of the leakage $L$ is at most $|L| \leq c \cdot n \log \ell$, for a fixed constant $c \in (0, 1)$.
- We consider the resulting conditional distribution $D$ on $R$ given leakage $L$.

**Theorem 3 (Informal).** Assume the set $R$ is drawn from a distribution $D \in C$. Then the min-hash protocol in the random oracle model preserves $(\epsilon, \delta)$-DDP for negligible $\delta$, as long as the size of the intersection is a constant fraction of the size of $A$ and $B$.

**On spoiling bits and leakage resilience.** Consider a distribution over sets of $n$ elements $R = R_1, \ldots, R_n$, where each $R_i$ is chosen from a universe of size $\ell \in O(n)$. Note that the set $R$ can have min-entropy $\Omega(n \log(\ell))$ while it can still be possible that for every $i$, the marginal distribution over $R_i$ has only constant min-entropy (see Example 1.1 in [20]). To deal with such situations, Skórski [40] proves a theorem showing the existence of “spoiling bits.” Namely, given $R_1, \ldots, R_n$, some additional information known as spoiling bits can be released such that, conditioned on this information, for each $i \in [n]$, the distribution of $R_i$ conditioned on $R_{<i}$, where $R_{<i}$ denotes $(R_1, \ldots, R_{i-1})$, is nearly flat (in the sense that the min/max entropy gap is at most a small additive constant). Further, the total number of spoiling bits that are released is small.

It is not hard to use Skórski’s result to show that if $R$ starts out with sufficiently high min-entropy then for a large fraction of $i$ (those in the set $V \subseteq [n]$), the distribution of $R_i$ conditioned on $R_{<i}$ has high min-entropy of at least $\Omega(\log(n))$, while the remaining indices (those in the set $W = [n] \setminus V$) may have low min-entropy.

Unfortunately, this result is very brittle in the sense that the flatness conditions hold only for this particular distribution of $R$ conditioned on the spoiled bits.
Specifically, despite the flatness condition being satisfied for this distribution, the random variables $R_i$ are not independent of one another. Thus, if additional information is leaked on $R_j$ after the spoiling bits are computed, then the flatness guarantees may no longer hold for $R_i$.

In our setting, we require additional leakage $\{\ell_i\}_{i \in W}$ on the elements $\{R_i\}_{i \in W}$. One issue is that the set $W$ (i.e., low min-entropy elements conditioned on the spoiling bits) is only known after the spoiling bits are computed. This leaves us with a dilemma:

- Leaking $\{\ell_i\}_{i \in W}$ additionally after the spoiling leakage can destroy the flatness property.
- On the other hand, if we want leak $\{\ell_i\}_{i \in W}$ before computing the spoiling bits, we are not able to do so, since we don’t know $W$ yet! We could leak from all the blocks $(R_1, \ldots, R_n)$, but then it may destroy the properties needed from the random variables $\{R_i\}_{i \in V}$.

To solve this problem, we prove a new variant of the spoiling lemma that computes the spoiling bits at the same time as the additional leakage $\ell_i$ for $i \in W$ is computed so that the spoiling bits also contain $\{\ell_i\}_{i \in W}$, while still maintaining the flatness condition. The types of leakage that can be captured are essentially those such that the leakage $\ell_i$ for $i \in W$ can be expressed as a function of $R_i$ and the leakages $\{\ell_j : j > i, j \in W\}$. It turns out that the leakage we need for our result has this form.

We state our theorem in general terms as we believe it may find further applications in leakage resilient cryptography. For the formal theorem statement see Theorem 4.

**Adversary model.** All of the protocols described above are secure in the random oracle model against a semi-honest adversary corrupting one of the parties. Achieving malicious security, especially for the DDP protocol, is an interesting open question.

**Empirical evaluation and reproducible computation.** We also perform empirical evaluation of our protocols corresponding to the first two theorems to determine the proper parameter ranges for privacy. While Theorem 3 obtains strictly better parameters than Theorem 2 from an asymptotic perspective, due to the constants behind the Big-O notation, the guarantees only kick in at large values of $n$. On the other hand, our empirical evaluation shows that Theorem 2 can be applicable for practical settings of $n$. We believe that the large constants in Theorem 3 are a byproduct of our analysis but are not inherent. Improving our analysis and optimizing the practical parameters are left as an open problem.

Our results show that in many cases, the min-hash sketch is already sufficient to achieve differential privacy of the inputs. This result allows us to build low-error and sublinear communication protocols for approximating the Jaccard index. Moreover, in addition to preserving privacy, these results are reproducible, generating the same output if run with the same inputs and the same hash functions.
Finally, this leads to the following encouraging observation. The min-hash has been in use for a long time, and in that time people have not really considered whether the results they were computing and storing were privacy preserving. As we now show that the min-hash sketch itself is DP, this gives evidence that prior computation may already preserve privacy of the individual inputs.

2 Related Works

**Differential privacy (DP).** Differential privacy protects the privacy of individuals by limiting an adversary’s ability to learn information about an individual input from the output of a computation \[16,18\].

A large body of works have developed differentially private algorithms \[19\] for a variety of computations. Most of these works focused on the standard setting with a trusted curator who has access to all users’ data and aims to respond in a differentially private manner.

Differential privacy has also been considered in a multi-party setting. In this setting two common approaches for differential privacy are local-DP, where parties add noise to their own inputs prior to performing the computation (e.g., \[21,31\]) or secure computation emulating the trusted curator (e.g., \[6\]).

**Optimizing secure computation using differential privacy.** Another direction of work has considered how to use DP to reduce the cost of secure computation, especially when we aim for DP-style guarantees from the final output. Beimel et al. \[6\] first proposed such optimization for the problem of secure summation. He et al. \[28\] and Groce et al. \[26\] applied the differential privacy relaxation to improve efficiency of set-intersection protocols. Mazloom and Gordon \[34\], and Mazloom et al. \[35\] consider graph-parallel computations and design more efficient solutions with differential private leakages. Chan et al. \[10\] consider classic tasks like sorting, merging, and range-query data structures with differential privacy relaxation. Gordon et al. \[25\] consider multiparty shuffle that allows a differentially private leakage and shows that it suffices to achieve end-to-end differential privacy in the shuffle model of DP.

**Private sketching.** Sketching algorithms, or "sketches" are sublinear space algorithms for approximating certain properties of large inputs or data streams. The main idea behind sketching algorithms is to generate a compact summary data structure that allows for efficient storage, merging, and processing. Some recent works \[7,12,42,44\] have additionally observed that sketches can often also aid in achieving privacy as the inherent loss of information in the sketch can essentially make the sketch itself be differentially private or to only require a little additional noise. As observed in e.g., \[12\] this observation about sketches can also lead to more efficient secure computation protocols.

**Secure Approximation.** Secure approximation studies what functions can be securely approximated without revealing anything beyond the true output \[23,27\]. While this notion is quite different from that of differentially-private approximation that we consider here, we note that our first (FHE-based) protocol additionally achieves this.
3 Preliminaries

Notations. A function $g$ is negligible, denoted $\text{negl}(\cdot)$, if for every positive integer $c$, there is an integer $n_c$ such that for all $n \geq n_c$ we have $g(n) \leq 1/n^c$. Let $\kappa$ denote the security parameter. Let $\mathcal{U}$ denote the universe of input elements. In this paper, we will consider two input sets $A, B \subseteq \mathcal{U}$. Let $n_A = |A|, n_B = |B|$. Let $I = A \cap B$, $n_I = |I|$. We will also let $B_{+x} = B \cup \{x^*\}$. Let $\text{Eq}$ be an equality function; i.e., $\text{Eq}(a, b) = 1$ if $a = b$ and 0 otherwise. For a hash function $h$ and a set $A$, we let $h(A) := \{h(a) : a \in A\}$. Let $\binom{m, p}$ be the binomial distribution with $m$ trials and each trial having success probability $p$.

Hash functions in the random oracle model. We model each hash function as a random oracle that maps each item to a real value in $[0, 1]$, and the output of the hash function is long enough to ensure that the probability of any two different items having a hash collision is $\text{negl}(\kappa)$.

Differential privacy. We first give the definition of the traditional $(\epsilon, \delta)$-differential privacy.

Definition 1 ((\(\epsilon, \delta\))-indistinguishability). Two random variables $X$ and $Y$ are $(\epsilon, \delta)$-indistinguishable (denoted as $X \approx_{\epsilon, \delta} Y$) if, for all events $S$, we have
\[
\Pr[X \in S] \leq e^\epsilon \cdot \Pr[Y \in S] + \delta, \quad \Pr[Y \in S] \leq e^\epsilon \cdot \Pr[X \in S] + \delta.
\]

Definition 2 (Computational (\(\epsilon, \delta\))-indistinguishability). Two random variables $X$ and $Y$ are computationally $(\epsilon, \delta)$-indistinguishable (denoted as $X \approx_{\epsilon, \delta} Y$) if, for any polynomial time adversary $A$, it holds
\[
\Pr[A(X) = 1] \leq e^\epsilon \cdot \Pr[A(Y) = 1] + \delta, \quad \Pr[A(Y) = 1] \leq e^\epsilon \cdot \Pr[A(X) = 1] + \delta.
\]

Definition 3 ((Computational) $(\epsilon, \delta)$-differential privacy). Let $X$ be an input space and $\simeq_X$ be a relation capturing the notion of neighboring inputs. Let $\mathcal{M} : X \rightarrow Z$ be a randomized algorithm that takes input $x \in X$ and outputs a value over $Z$. We say that the mechanism $\mathcal{M}$ is $(\epsilon, \delta)$-differentially private if the following holds:
\[
\forall x, x' \in X \text{ s.t. } x \simeq_X x' : \mathcal{M}(x) \approx_{\epsilon, \delta} \mathcal{M}(x').
\]
The mechanism $\mathcal{M}$ is $(\epsilon, \delta)$-computationally differentially private if $\forall x, x' \in X \text{ s.t. } x \simeq_X x' : \mathcal{M}(x) \approx_{\epsilon, \delta} \mathcal{M}(x')$.

Distributional differential privacy (DDP). We adapt the original definition [3] for our purpose to consider a two-party protocol that takes sets as input more explicitly. Specifically, we consider the following computational indistinguishability variant for our DDP definition.

Definition 4 (View of a party in a two-party protocol). Given a two-party protocol $\Pi$ with parties $P_1$ and $P_2$, let $\text{view}_{P_1}^\Pi(A, B)$ denote the view of $P_1$ for the execution of protocol $\Pi$ with $A$ and $B$ being the input of $P_1$ and $P_2$ respectively. In particular, $\text{view}_{P_1}^\Pi(A, B)$ consists of the following:
The input $A$ of $P_1$, the randomness that $P_1$ uses, the messages that $P_1$ receives from $P_2$, and the output of the protocol.

If the protocol is in the random oracle model, we allow a semi-honest $P_1$ to make a polynomial number of arbitrary queries to the random oracle and to add the input/output information to its view.

The view of $P_2$ is defined similarly.

We now define computational DDP for a two-party protocol against an adversary that can corrupt one of them in an honest-but-curious manner.

**Definition 5 (Computational DDP of a two-party protocol).** Let $\mathcal{X}$ denote a random variable for two sets over universe $\mathcal{U}$. Let $Z$ denote the random variable measuring the additional auxiliary information known to the adversary. A two party protocol $\Pi$ is computationally $(\epsilon, \delta, \Delta)$-DDP against an adversary corrupting $P_1$, if for every distribution $D \in \Delta$ on $(\mathcal{X}, Z)$, every $(X = (A, B), Z)$ in the support of $(\mathcal{X}, Z)$ and every $x^* \in \mathcal{U}$, the following holds:

$$\text{view}^R_{\Pi}(A, B, Z) \approx^{\epsilon, \delta} \text{view}^R_{\Pi}(A, B + x^*), Z).$$

In the above, $(A, B)$ and $Z$ are sampled from $\Delta$, and each party may use additional randomness. DDP against an adversary corrupting $P_2$ is defined symmetrically.

Finally, the following lemma upperbounds the tail of a Binomial distribution.

**Lemma 1 ([15]).** Consider a Binomial distribution $B(n, p)$. We have

$$\Pr_{X \sim B(n, p)}[X \geq k] \leq \binom{n}{k} p^k.$$

### 4 Our Protocols

#### 4.1 Min-Hash Protocol in the Trusted Curator Model

In Figure 1, we describe the standard min-hash protocol in the trusted curator model ($F_{cu}$). In particular, after choosing $k$ random hash functions, the mechanism computes the number of iterations in which the min-hash of $A$ matches the min-hash of $B$.

**Theorem 1.** For any constant $\epsilon > 0$, if $k = k(\epsilon, \kappa) \in \Omega(\kappa), n_A/k \in \Omega(\kappa), n_B/k \in \Omega(\kappa)$, and $J(A, B) \in (0, 1)$ is a constant independent of $\kappa$, then $F_{cu}$ is $(\epsilon, \delta)$-DP with $\delta \in \text{negl}(\kappa)$.

While $F_{cu}$ is described in the trusted curator model, a two-party protocol realizing it can be constructed without relying on a trusted curator. In particular, the computation of $(u^A_1, \ldots, u^A_k)$ (including all $n$ hash evaluations) can be performed locally under a (threshold) FHE so that only the encryption of them may be sent to party $B$. Then, by computing the remaining steps under FHE and delivering the result using a threshold decryption, the protocol will securely realize $F_{cu}$ in the semi-honest setting. We note that the resulting protocol has sublinear communication in $n$ since only the $k$ inputs to the comparisons need to be communicated.
Min-Hash in the Trusted Curator Model $\mathcal{F}_{\text{cu}}$

**Input:** $P_1$ and $P_2$’s input vectors $A = (x_1^A, \ldots, x_n^A)$ and $B = (x_1^B, \ldots, x_n^B)$.

**Minhash:**

1. Randomly sample $k$ hash functions $h_1, h_2, \ldots, h_k$.
2. For input $A$, compute the min-hash vector $(u_1^A, u_2^A, \ldots, u_k^A)$ as follows:
   
   i. For each item $x_i^A \in A$, compute $y_{i,j}^A = h_j(x_i^A)$.
   
   ii. Compute the min-hash for iteration $j$; that is, $u_j^A = \min_i\{y_{i,j}^A\}$
3. Likewise, compute another min-hash vector $(u_1^B, u_2^B, \ldots, u_k^B)$ for input $B$ similarly.
4. Compute $c = \sum_{j=1}^k \text{Eq}(u_j^A, u_j^B)$.

**Output:** Return $c$ to $P_1$ and $P_2$.

Fig. 1: Min-Hash in the Trusted Curator Model

### 4.2 Two-party Min-Hash Protocol with DDP

In Figure 2, we describe a two-party protocol $\pi^{O}_{\text{PH}}$ in the random oracle (RO) model achieving DDP. The protocol is essentially the same as $\mathcal{F}_{\text{cu}}$, except for the following differences:

- Prefixes $\text{pre}_1, \ldots, \text{pre}_k$ are jointly sampled by the parties.
- The hash functions $h_i$ for $i \in [k]$ are defined as $h_i(\cdot) := O(\text{pre}_i||\cdot)$.
- A generic secure two-party computation [48,33] is used to hide the information of $u_i^A$’s and $u_i^B$’s in the equality check.
- The parties’ output consists of the min-hash output, as well as the pre-fixes $\text{pre}_1, \ldots, \text{pre}_k$. This additional information greatly impacts the analysis.

The main benefit of the protocol is that the hash computations can be computed locally in the clear without FHE.

#### Distributional Differential Privacy

In Figure 3, we describe the family of distributions we consider in the context of our min-hash protocol. The distribution models a situation in which the adversary, having corrupted one of the two parties, has access to the view of the party and even the actual intersection. However, the adversary does not know the other party’s input set (except from the intersection).

We assume that each of the non-intersecting elements has high min-entropy. WLOG, consider an adversary corrupting $P_1$. We can safely ignore the protocol messages in Step 3 in our analysis, thanks to the security guarantees of secure two-party computation. Therefore, the view of the adversary will be

$$\text{view}^\text{ph}_{P_1}(A, B) := (c, h_1, \ldots, h_k).$$
Protocol $\pi^D_{\text{PH}}$

**Input:** $P_1$ and $P_2$’s input vectors $A = (x^A_1, \ldots, x^A_n)$ and $B = (x^B_1, \ldots, x^B_n)$.

**MinHash:**

1. Both parties jointly sample $k$ prefixes $\text{pre}_1, \ldots, \text{pre}_k$. These are used to define hash functions $h_1, h_2, \ldots, h_k$, where for $i \in [k]$, $h_i(\cdot) := O(\text{pre}_i||\cdot)$.
2. Party 1 takes its input $A$ from which it computes the min-hash vector $(u^A_1, u^A_2, \ldots, u^A_k)$ exactly as described in $F_{\text{cu}}$. Likewise, Party 2 computes $(u^B_1, u^B_2, \ldots, u^B_k)$ similarly using its input $B$.
3. Taking $(u^A_1, u^A_2, \ldots, u^A_k)$ and $(u^B_1, u^B_2, \ldots, u^B_k)$ as input, both parties execute a generic secure-two party computation protocol for computing $c := \sum_{j=1}^{k} \text{Eq}(u^A_j, u^B_j)$.

**Output:** Return $c$, $\text{pre}_1, \ldots, \text{pre}_k$, to $P_1$ and $P_2$.

---

**Fig. 2:** The Protocol for Min Hash in the RO Model

**Distribution Family $\Delta_{\text{PH}}$**

This family of distributions is parameterized as follows:

- The universe $\mathcal{U}$ for the elements in sets $A$ and $B$; the cardinalities $n_A, n_B, n_I$ of sets $A$, $B$, $I = A \cap B$ respectively; distribution $\mathcal{D}_{A,I,B'}$ for sampling $(A, I)$ and $B' = B \setminus A$.

Choose $(A, I, B')$ according to $\mathcal{D}_{A,I,B'}$, which should satisfy the following requirements:

- $|A| = n_A, |I| = n_I, |B'| = n_B - n_I, I \subset A, A \cap B' = \emptyset$.

**Output:**

- The inputs to the parties $P_1$ and $P_2$ are $A$ and $B$ respectively.
- Give $I$ to the adversary as the auxiliary information.

---

**Fig. 3:** The family of distributions that we consider in our min-hash protocol

As with the case of $F_{\text{cu}}$, since the hash functions are chosen at random, we can apply the same analysis and show that the sensitivity can be upper-bounded by a small value $s$. Unlike $F_{\text{cu}}$, however, when we show the existence of sufficient noise from the remaining iterations, we need to take the additional leakage into consideration.

**Hashes of non-intersecting items working as a noise.** First, since the hash functions are public, iterations are no longer independent of each other as needed by the analysis in Section 4.1. We address this issue by employing the fact that each of the non-intersecting items has high min-entropy. In the random oracle model, as long as the adversary does not query hash function $h$ on some
point $x$, $h(x)$ is uniformly random to the adversary. Since the non-intersecting items have high min-entropy, the adversary is negligibly likely to query any of them to the hash functions, thus guaranteeing independence.

Now, to see how the remaining iterations still hide the sensitivity even with the public hash functions, let $R = B \setminus I$. For the remaining $k - s$ iterations, the high min-entropy of each element in $R$ will jitter the final count. In particular, consider the $j$th hash function $h_j$ in the protocol (among the $k - s$ remaining iterations) and let

$$v_j^A = \min h_j(A), \quad v_j^I = \min h_j(I), \quad v_j^R = \min h_j(R).$$

Suppose $v_j^A = v_j^I$. Then, if $v_j^R \geq v_j^I$, the min-hash $u_j^A$ of $A$ will be equal to the min-hash $u_j^B$ of $B$ (both of which are equal to $v_j^I$) and the final count $c$ will be incremented due to this $j$th iteration. However, if $v_j^R < v_j^I$, then it will be $u_j^A \neq u_j^B$, and the final count will not be incremented. This way, the distribution of $v_j^R$ will jitter the final count. Let $n_R = n_B - n_I$, and the above discussion can be formalized into the following definition.

**Definition 6 ($\theta$-good iteration).** We define a predicate $\text{good}_\theta(h_j, A, I, n_B)$ to be true if and only if the following holds:

$$\min h_j(A) = \min h_j(I), \quad \min h_j(I) \in \left[1 - \left(\frac{1}{2} + \theta\right)^{1/n_R}, 1 - \left(\frac{1}{2} - \theta\right)^{1/n_R}\right].$$

The second condition of the definition requires that $\min h_j(I)$ is somewhere in the middle (parameterized by $\theta \in \Theta(1)$) so that $v_j^R$ may reduce the final count with a decent chance, and it may not with a decent chance too. As long as $n_I/n_A$ is a constant fraction, there are sufficiently many $\theta$-good iterations, although we lose some iterations. In particular, if we let $k_g$ be the number of good iterations, we have $k_g = \Theta(k)$.

Since the hash functions are now public, and therefore $\min h_j(I)$ is leaked to the adversary, it turns out that the noise from the $k_g$ iterations follows a Poisson Binomial distribution, which is a generalization of a Binomial distribution where each trial has a different success probability. However, we can still show that this distribution works as a good noise to hide the private data. In particular, we use the techniques of [11] to upper-bound the privacy guarantee as those from a binomial mechanism (with rather worse parameters compared to $\mathcal{F}_k$).

**Theorem 2.** For every constant $\epsilon > 0$, consider protocol $\pi_{PH}$ in the random oracle model with $k = k(\epsilon, \kappa)$, where $k \in \Omega(\kappa)$. Let $R = B \setminus I$, each element of which has min-entropy at least $\kappa$. Let $n_A/k, n_B/k \in \Omega(\kappa)$, and $n_I/n_A \leq (0, 1)$ is a constant independent of $\kappa$. Then, protocol $\pi_{PH}$ is computationally $(\epsilon, \delta, \Delta_{PH})$-DDP against an adversary corrupting $P_1$ with $\delta \in \text{negl}(\kappa)$. DDP against an adversary corrupting $P_2$ holds when the parameters are set symmetrically.
4.3 DDP for Input Sets Chosen from a Poly-sized Universe

Let \( n_R = n_B - n_I \). In this section, we show that \( \pi_{PH} \) satisfies DDP even when the size of the universe \( \mathcal{U} \) of size \( n_R \cdot \ell \) is polynomial in \( \kappa \) with \( \ell = \Omega(n_R^3) \), and the secret set \( R \) is chosen from the uniform distribution on \( \mathcal{U} \), conditioned on arbitrary leakage on \( R \) of length \( L \), where \( L \leq n_R(\lg \ell - 3 \lg n_R - 2) \). Assume without loss of generality that the adversary corrupts \( P_1 \).

**Noise: fixed secret set over the choice of hashes.** Consider a fixed secret set \( R' \subset R = B \setminus I \) (we will see how to choose \( R' \) later). Let \( n = |R| \) and \( n' = |R'| \) and consider the probability \( E_r^{R'} \) over choice of the hash functions that \( R' \) contributes noise of \( -r \) to the total count \( c \) in the min-hash protocol over a bundle of \( k_b \) good iterations (we will see how to choose \( k_b \) later). We can show that the distribution over \( r \) has the following property: There exist \( a, b \in \{0, \ldots, k_b\} \) such that:

- The probability of obtaining \( r \notin [a, b] \) is negligible.
- For every \( \epsilon \in [0, 1] \) and sufficiently large security parameter \( \kappa \), for every \( r \in [a, b] \), we have \( e^{-\epsilon} \leq \frac{\Pr[r]}{\Pr[r+E_r]} \leq e^{\epsilon} \) where \( s = \lg \lg \kappa \) is a small value corresponding to the sensitivity.

The above indicates that the distribution over \( r \), when the set \( R' \) is fixed and the probability is taken over choice of hash functions, is amenable for use as a noise distribution in a differential privacy context. It turns out \( E_r^{R'} \) depends on only the size of \( R' \) and for every \( r \), we let \( E_r^{n'} \) denote its probability under the distribution described above.

**Noise: fixed hash over the choice of the secret set.** Our main technical challenge is to show that the properties needed for differential privacy hold even when we switch quantifiers. Specifically, given a distribution \( \mathcal{D} \) over sets \( R' \) and a fixed hash function (modeled as a random oracle), let \( d_r \) be the probability over \( R' \) chosen from \( \mathcal{D} \) that \( R' \) contributes \( -r \) to the total count in the min-hash protocol. We would like to show that for any fixed hash functions, for all \( r \in [a, b] \), it holds \( e^{-\epsilon} E_r^{n'} \leq d_r \leq e^{\epsilon} E_r^{n'} \). Then, we can meaningfully upper-bound the distance between \( d_r \) and \( d_{r+E_r} \) using the property of \( E_r^{n'} \) described above.

The aforementioned universal quantifier for the hash functions can be slightly relaxed by requiring the property to hold over the choice of the hash functions with all but small probability (we will later describe how to compensate the relaxation by the bundling technique). In particular, let \( D_r \) be the random variable corresponding to probability \( d_r \) over the choice of hash functions. We would like to show that with all but small probability over the choice of the hash functions, for all \( r \in [a, b] \), it holds \( e^{-\epsilon} E_r^{n'} \leq D_r \leq e^{\epsilon} E_r^{n'} \).

**Geometric collision Property.** To show the above, our high-level strategy is to use Chebyshev’s inequality, which gives good concentration bounds on \( D_r \) when \( \Var[D_r] \) is bounded. Thus, our next goal is to upperbound \( \Var[D_r] \). To do so, we introduce a property of distributions \( \mathcal{D} \) over sets \( R' \) which we call the “Geometric Collision Property”. In a nutshell, this property states that the probability that two sets \( R'_1, R'_2 \) drawn independently from \( \mathcal{D} \) have intersection
of size $z$ is at most $(\frac{1}{n})^z$ for all $z \in [n']$. We show that $\text{Var}[D_r]$ can be bounded for any distribution over sets $R'$ that has this property.

**Geometric collision property in the face of leakage.** It is not hard to see that the uniform distribution over all sets $R'$ of size $n'$ from a universe of size $n' \cdot \ell$ (where $\ell \in \Omega(n^3)$) satisfies the “Geometric Collision Property”. It would seem, therefore, that we could take this as our secret distribution and the analysis would be complete. Unfortunately, even for the case in which the distribution is sets of size $n'$ chosen uniformly at random from the universe, the analysis is not straightforward. The difficulty stems from the fact that the “noise” in the protocol is tied to the input itself. Therefore, if information about the input is leaked in any other part of the protocol, then the noise distribution changes and may no longer satisfy the required properties. Specifically in our case learning the number of matches across the two parties’ sets with respect to some of the hash functions leaks information about the secret set of the honest party (since the secret set affects those counts).

**Strong chain-rule for min-entropy.** We first observe that our initial min-entropy in the distribution over secret sets $R$ is high (approximately $\frac{8n}{9} \lg \ell + 2n$) and that the entire information leaked about $R$ from the counts of the iterations that are not $\theta$-good is small. We can lower-bound the remaining min-entropy in $D$, therefore, using the weak chain rule for min-entropy [14, Lemma 2.2]; if we want to lower bound the remaining min-entropy with all but $2^{-\kappa}$ probability, however, we need to take a hit of $\kappa$ in the min-entropy.

Recall that each individual element in $R$ can be viewed as being chosen from a set of size $\ell$ and thus has min-entropy of at most $\lg(\ell) \ll \kappa$. Thus, after applying the weak chain rule and losing more than $\kappa$ bits of min-entropy, we can have certain elements that have only constant min-entropy, thus implying that collisions are likely in those positions. So the weak chain rule, while leaking only a small number of bits overall, can ruin the geometric collision property. Even worse, the min-entropy definition doesn’t rule out the case in which all elements of $R$ (i.e. the marginal distributions over each element in $R$) have only constant min-entropy, while the total min-entropy in $R$ remains high!

This phenomenon has been previously observed and studied in the literature [41]. One way to deal with such a counter-intuitive situation is to actually leak a small amount of *additional* information, known as “spoiled” bits. This will lower the total min-entropy in $R$, but will ensure that a large fraction of blocks in $R$ still have high min entropy of at least $1.5 \lg(n)$. We extend the techniques of [41] to produce spoiling leakage with the following properties:

- It can be computed element-by-element, starting from the last element of the distribution.
- Conditioned on the spoiling leakage, we can simulate the adversary’s entire view in the protocol (by leaking extra information along with the spoiled bits), other than the outcome of the random variable $r$ over choice of $R$, where $R$ is drawn from the probability distribution that conditions on the spoiling leakage.
– Conditioned on the spoiling leakage, each element $R_i$ either has max-entropy at most $\sim 1.5 \cdot \lg(n)$ or min-entropy at least $\sim 1.5 \cdot \lg(n)$.

We further show that, due to the high min-entropy of the entire set $R$, there must be a constant fraction of blocks $R'$ with min-entropy at least $1.5 \cdot \lg(n)$. To reflect the fact that only a constant fraction of elements are guaranteed to have high min-entropy, the parameter $n'$ is set to $n/3$. We now consider the marginal distribution over $R'$ (the elements of $R$ with high min-entropy) and show that the geometric collision property holds with respect to this distribution.

**Bundling iterations towards DDP with negligible $\delta$.** We are not quite done yet. By applying Chebyshev we are only able to reduce the failure probability only to $\sim 1/\sqrt{n}$, whereas we would like the failure probability to be negligible. In order to do that, we split the “good” iterations into $u$ bundles, where $u$ is a small superconstant number, and argue that w.h.p. Chebyshev succeeds for at least one bundle. Note that the hash functions are independent in each bundle and so the probability that all $u$ bundles fail should be $(\frac{1}{\sqrt{n}})^u$, which is negligible for superconstant $u$. For this, we set the parameter $k_b = k_j/u$, where $k_j$ is the number of good iterations. The fact that the probability distribution of $r$ over choice of $R$ is good for at least one bundle (with all but negligible probability) is sufficient to complete the proof of distributional differential privacy.

**Theorem 3.** For security parameter $\kappa$, every constant $\epsilon > 0$, and every constant $\gamma \in (0, 1)$, consider protocol $\pi_{\text{PH}}$ in Figure 2 in the random oracle model with $k = k(\epsilon, \kappa)$, where $k \in \Omega(\kappa \cdot \lg \lg \kappa)$. Let $R = B \backslash I$ be a set of size $n_R$, with $n_R/k^2 \in \Omega(\kappa)$. Let the universe $U$ be of size $n_R \cdot \ell$, where $\ell = \Omega(n_R^3)$. Assume the secret set $R$ is chosen chosen uniformly from all subsets of $U$ of size $n_R$, conditioned on arbitrary leakage on $R$ of length $L$, where $n_R \lg \ell - L \geq \frac{8n_R^2}{9} \lg \ell + 2n_R$. Let $|I| \in \Theta(n)$. Then the output of $\pi_{\text{PH}}$ achieves computational $(\epsilon, \delta, \Delta_{\text{PH}})$-DP with $\delta \in \text{neg}(\kappa)$ against an adversary corrupting $P_1$. DDP against an adversary corrupting $P_2$ holds when the parameters are set symmetrically.

## 5 DP of $\mathcal{F}_{ca}$: Proof of Theorem 1

**Setting the scene.** WLOG, we consider two neighboring inputs

$$(A, B) \text{ and } (A, B_{+x^*}).$$

Differential privacy for the case in which $x^*$ is added into $A$ can be shown symmetrically. WLOG, let $B = (x_1^B, \ldots, x_n^B)$ and $B_{+x^*} = (x_1^B, \ldots, x_{n_B}^B, x^*)$.

**Sensitivity.** We show how changing the input sets from $B$ to $B_{+x^*}$ affects the final count. Let $x^*$ be the $(n_B + 1)$-th element of $B_{+x^*}$. Consider iteration $j$. Since we model each hash function $h_j$ as a random oracle, $(y_{1,j}^B, \ldots, y_{n_B+1,j}^B)$ will be uniformly distributed. We observe the following:

Consider how the min-hash $v_j^B$ is computed. The value $x^*$ from $B_{+x^*}$ can affect the min-hash $u_j^B$ (and thereby the final count $c$), only if $y_{n_B+1,j}^B$ is smaller than $(y_{1,j}^B, \ldots, y_{n_B,j}^B)$.
The probability that $y_{n_B+1,j}^B$ will be the minimum is $1/(n_B + 1)$ by a symmetry argument. Note the final output is computed as the sum of $k$ of these trials. Let

$$S_{x^*} = \left\{ j \in [k] : y_{n_B+1,j}^B = \min_{i \in [n_B+1]} \{ y_{i,j}^B \} \right\}.$$ 

Therefore, we consider a binomial distribution as follows

$$|S_{x^*}| \sim B(k, 1/(n_B + 1)),$$

which represents how many iterations $j$ cause $x^*$ to be the min-hash $u_j^B$. In other words, $|S_{x^*}|$ captures the sensitivity of min-hash.

The following lemma upper bounds this sensitivity.

**Lemma 2.** For any $\{x_i^B\}_{i \in [n_B]}$ and any $x \in U$, we have $\Pr_{h_1,\ldots,h_k}[|S_x| \geq s] \leq \left(\frac{e^{-\epsilon k}}{\epsilon}\right)^s \cdot \left(\frac{e \cdot \frac{1}{n_B+1}}{s}\right)^s \cdot \left(\frac{e k}{s}\right)^s \cdot \left(\frac{1}{n_B+1}\right)^s$. The first inequality is from Lemma 1.

**Proof.** We have $\Pr[|S_x| \geq s] \leq \left(\frac{k}{n_B+1}\right)^s \cdot \left(\frac{1}{n_B+1}\right)^s \leq \left(\frac{e k}{s}\right)^s \cdot \left(\frac{1}{n_B+1}\right)^s$. The first inequality is from Lemma 1.

**Remark.** From the above lemma, for $k \in \Theta(\kappa)$, $n_B \in \Omega(\kappa^2)$, we have

$$\Pr[|S_{x^*}| \geq \lg \lg \kappa] \leq \negl(\kappa).$$

This implies that given the parameters above, with overwhelming probability, there are at most $\lg \lg \kappa$ iterations where $x^*$ changes the min-hash value for $B$.

**Binomial distribution hides the small sensitivity.** Let $s = \lg \lg \kappa$ and let $p = J(A, B)$. Let $K_{x^*} = [k] \setminus S_{x^*}$. For the iterations in $K_{x^*}$, the min-hash matches for both $(A, B)$ and $(A, B^+_{x^*})$ will be identically distributed.

Let $c_{K_{x^*}}$ denote the match count in iterations $K_{x^*}$. Note that since we model each hash function as a random oracle, we have

$$c_{K_{x^*}} \sim B(k - s, p).$$

By applying Lemma 3 below, we conclude that $F_{eu}$ is differentially private.

**Lemma 3.** Consider a Binomial distribution $B(n, p)$, where $n \in \Omega(\kappa)$ and $p \in (0, 1)$ is a constant independent of $\kappa$. Then, for any constant $\epsilon$ and $s \leq \lg \lg \kappa$, there are $a, b \in [n]$ with $a < np < b$ such that

- For any $\ell \in [a, b]$, $e^{-\epsilon} \leq \frac{\Pr[B(n, p) = \ell]}{\Pr[B(n, p) + s = \ell]} \leq e^\epsilon$.  
- For any $\ell \notin [a, b]$, $\Pr[B(n, p) = \ell] = \negl(\kappa)$ and $\Pr[B(n, p) + s = \ell] = \negl(\kappa)$. 

16
WLOG, we consider two neighboring inputs \((A, B)\) and \((A, B_{+x^*})\). DDP for the case in which \(x^*\) is added into \(A\) can be shown symmetrically.

We prove the theorem by a hybrid argument. We first define the following ideal functionality \(F_{\text{PH}}^0\) in the random oracle (RO) model.

We set up the following hybrids. We will argue that for any \(x^* \in U\) and over \((A, B, I) \leftarrow \Delta_{\text{PH}}, 1\) it holds

\[
\text{view}_{P_1}^{\text{PH}}(A, B, I) \approx_c \text{view}_{P_1}^{\text{PH}}(A, B, I) \approx (\text{view}_{P_1}^{\text{PH}}(A, B), I)
\]

for any constant \(\epsilon > 0\) and for some \(\delta \in \text{negl}(\kappa)\), as long as each element in \(B \setminus I\) has high min-entropy.

Thanks to the standard technique of the generic two-party secure computation \[33\], the following holds:

\[
\text{view}_{P_1}^{\text{PH}}(A, B, I) \approx_c (\text{view}_{P_1}^{\text{PH}}(A, B), I, \text{view}_{P_1}^{\text{PH}}(A, B_{+x^*}), I) \approx (\text{view}_{P_1}^{\text{PH}}(A, B), I).
\]

Now we show \((\text{view}_{P_1}^{\text{PH}}(A, B), I) \approx \text{view}_{P_1}^{\text{PH}}(A, B), I)\). Recall that the min-entropy of each element \(x_i^B\) with \(i \in B \setminus A\) is at least \(\kappa\). Therefore, the probability that any adversary making at most polynomially many oracle queries queries any \(x_i^B\) is \(\text{negl}(\kappa)\). Conditioned on the adversary not querying any such \(x_i^B\), any \(y_{i,j}^B\) for \(j \in [k]\) is chosen uniformly random from \(U\). The same argument shows

\[
\text{view}_{P_1}^{\text{PH}}(A, B_{+x^*}, I) \approx (\text{view}_{P_1}^{\text{PH}}(A, B_{+x^*}, I)).
\]

### 6 DDP of \(\pi_{\text{PH}}\): Proof of Theorem 2

We also define a slightly different ideal functionality \(F_{\text{PH}}^{(1)}\) as follows:

- Let \(F_{\text{PH}}^{(1)}\) be the same as \(F_{\text{PH}}\) except that for each \(x_i^B \in B \setminus A\), each element in \(\{y_{i,j}^B\}_{j}\) is chosen uniformly at random from \([0, 1]\).

We set up the following hybrids. We will argue that for any \(x^* \in U\) and over \((A, B, I) \leftarrow \Delta_{\text{PH}}, 1\) it holds

\[
\text{view}_{P_1}^{\text{PH}}(A, B, I) \approx_c (\text{view}_{P_1}^{\text{PH}}(A, B), I) \approx (\text{view}_{P_1}^{\text{PH}}(A, B), I)
\]

for any constant \(\epsilon > 0\) and for some \(\delta \in \text{negl}(\kappa)\), as long as each element in \(B \setminus I\) has high min-entropy.

Thanks to the standard technique of the generic two-party secure computation \[33\], the following holds:

\[
\text{view}_{P_1}^{\text{PH}}(A, B, I) \approx_c (\text{view}_{P_1}^{\text{PH}}(A, B), I, \text{view}_{P_1}^{\text{PH}}(A, B_{+x^*}), I) \approx (\text{view}_{P_1}^{\text{PH}}(A, B), I).
\]

Now we show \((\text{view}_{P_1}^{\text{PH}}(A, B), I) \approx \text{view}_{P_1}^{\text{PH}}(A, B), I)\). Recall that the min-entropy of each element \(x_i^B\) with \(i \in B \setminus A\) is at least \(\kappa\). Therefore, the probability that any adversary making at most polynomially many oracle queries queries any \(x_i^B\) is \(\text{negl}(\kappa)\). Conditioned on the adversary not querying any such \(x_i^B\), any \(y_{i,j}^B\) for \(j \in [k]\) is chosen uniformly random from \(U\). The same argument shows

\[
\text{view}_{P_1}^{\text{PH}}(A, B_{+x^*}, I) \approx (\text{view}_{P_1}^{\text{PH}}(A, B_{+x^*}, I)).
\]

### 6.1 DDP of \(F_{\text{PH}}^{(1)}\)

From the above discussion, we are only left to show

\[
(\text{view}_{P_1}^{\text{PH}}(A, B), I) \approx_{x, \delta} (\text{view}_{P_1}^{\text{PH}}(A, B_{+x^*}), I).
\]
In other words, we need to show
\[(A, I, h_1, \ldots, h_k, c) \approx_{\epsilon, \delta} (A, I, h_1, \ldots, h_k, c + x^*)\]
where \(c\) is the final count from \(F_{PH}^{(1)}(A, B)\) and \(c + x^*\) is the final count from \(F_{PH}^{(1)}(A, B + x)\).

In Section 5, we studied the sensitivity of the final count when adding an element \(x^*\) to \(B\). We show how to leverage the uncertainties of \(x^*\) so that good iterations work like the needed noise to guarantee DP.

The following lemma shows that in the random oracle model, a random hash leads to a good iteration with probability \(p_{\theta}\), which is constant in our setting based on the assumption about \(n_A, n_I, n_R\).

**Lemma 4.** For any \(A, I, n_B\) and \(n_R = n_B - |I|\), we have
\[
p_{\theta} \overset{\text{def}}{=} \Pr_h [\text{good}_{\theta}(h, A, I, n_B)] \geq \left( \left( \frac{1}{2} + \theta \right)^{\frac{n_A}{n_R}} - \left( \frac{1}{2} - \theta \right)^{\frac{n_A}{n_R}} \right) \cdot \frac{n_I}{n_R}
\]

Recall that \(S_{x^*}\) was the random variable that represents the set of iterations \(j\) such that the min-hash \(u^B_j\) comes from \(x^*\) when \(P_2^2\)'s input is \(B + x^*\). From Lemma 2, with overwhelming probability \(|S_{x^*}| \leq \log \log \kappa\).

Now, let \(G_\theta\) be the set of iterations \(j\) in which a \(\theta\)-good event takes place; i.e.,
\[
G_\theta = \{ j \in [k] : \text{good}_{\theta}(h_j, A, I, n_B) \}.
\]

Let \(K_\theta = G_\theta \setminus S_{x^*}\). The following lemma shows that the \(\theta\)-good events takes place \(\Theta(\kappa)\)-many times, with overwhelming probability.

**Lemma 5.** Suppose \(k = \Theta(\kappa)\), \(n_B = \Omega(\kappa^2)\), and \(p_{\theta} \in \Theta(1)\). Let \(s = |S_{x^*}|\). Then, we have
\[
\Pr_{h_1, \ldots, h_k} [ |K_\theta| > \frac{2}{3} (k - s) p_\theta ] \geq 1 - \negl(\kappa).
\]

**Fixing randomness.** Recall we aim to show that
\[(A, I, h_1, \ldots, h_k, c) \approx_{\epsilon, \delta} (A, I, h_1, \ldots, h_k, c + x^*)\]  \hspace{1cm} (1)

To achieve this goal, we divide the output count into two parts:

- The part corresponding to \(K_\theta\), which contains all the \(\theta\)-good iterations such that \(x^*\) does not hash to the minimum across \(B + x^*\).
- The part contributed by all the remaining iterations.

We will first argue that the difference of the second parts for the neighboring inputs \(B\) and \(B + x^*\) is upper-bounded by a small value, following our discussion on sensitivity in Section 5. Then we will treat the first part as the noise distribution and derive the privacy guarantee it provides to hide the difference from the second part.

For a set \(W \subset [k]\), define its complement as \(\overline{W} = [k] \setminus W\). In addition, we introduce the following notations.
For sets $V, W$, let $Y_{V, W} \overset{def}{=} \{y_{i,j}^B : i \in V, j \in W\}$.

For a set $W$, define $c_W \overset{def}{=} \sum_{j \in W} \text{Eq}(u_j^A, u_j^B)$ and 

Now, fix $A, I, x^*$ and $h_1, \ldots, h_k$. Recall that we are considering $F^{(1)}_{PH}$ that satisfy the following condition:

- For each $x_i^B \in B \setminus A$, each element in $\{y_{i,j}^B\}_j$ is chosen uniformly at random from $[0, 1]$.

Therefore, even after $(A, I, x^*, h_1, \ldots, h_k)$ is fixed, it holds that $Y_{R,[k]}$ are still uniformly distributed. Moreover, we additionally fix $Y_{R,G\theta}$, which implies that only $Y_{R,G\theta}$ is only uniformly random.

Note that fixing $A, I, x^*, h_1, \ldots, h_k$ also allows us to determine which iterations are $\theta$-good (i.e., $G\theta$).

**Small sensitivity.** Given all this, we first show that there for $s = \lg \lg \kappa$,

$$\Pr_{Y_{R,G\theta}} \left[ |c_{K\theta} - c^{x^*}_{K\theta}| > s \right] \leq \text{negl}(\kappa).$$

To show inequality, noting that $K\theta = G\theta \cup S_{x^*}$, consider two cases:

- For iteration $j \in G\theta \setminus S_{x^*}$, we have $u_j^B = u_j^{B+x^*}$. This is because $j \notin S_{x^*}$ implies that $x^*$ doesn’t lead to min-hash. Therefore, such iterations don’t contribute to the difference of $c$ and $c^{x^*}$.
- For the rest iterations $j \in S_{x^*}$, we have $|S_{x^*}| \leq s$ with overwhelming probability due to Lemma 2, which shows the above equation.

**Our goal.** Essentially, for any final count $q$, we are interested in comparing the two probabilities:

$$\Pr_{Y_{R,G\theta}} [c_{K\theta} + c_{K\theta} = q] \text{ and } \Pr_{Y_{R,G\theta}} [c^{x^*}_{K\theta} + c^{x^*}_{K\theta} = q].$$

Note that we have $c_{K\theta} = c^{x^*}_{K\theta}$ because $j \in K\theta$ implies $j \notin S_{x^*}$. Therefore, based on the sensitivity argument shown above, we only need to analyze the distribution of $c_{K\theta}$ and compare the following two probabilities:

$$\Pr_{Y_{R,G\theta}} [c_{K\theta} = q] \text{ and } \Pr_{Y_{R,G\theta}} [c_{K\theta} + s = q].$$

**Distribution of $c_{K\theta}$.** We abuse notation and denote $c_j = c_{(j)}$. Then, we have $c_{K\theta} = \sum_{j \in K\theta} c_j$. Let $Y_{R,j} = \{y_{i,j}^B : i \in (n_I, n_B)\}$. Note that since we have $j \in K\theta$, a $\theta$-good event takes place in iteration $j$, i.e., $\min h_j(A) = \min h_j(I)$. This implies the following:

If $\min Y_{R,j} \geq \min h_j(I)$, we have $c_j = 1$; otherwise we have $c_j = 0$.  

19
Let $\gamma_j = 1 - \min h_j(I)$. Then, the probability that $c_j = 1$ is $(\gamma_j)^{n_R}$, since every number in $Y_{R,j}$ must be be greater than or equal to $\min h_j(I)$. Let $\eta_{-\theta} = 1/2 - \theta$ and $\eta_{+\theta} = 1/2 + \theta$. Note that since $j \in G_\theta$, the following holds according to Definition 8:

$$(\gamma_j)^{n_R} \in [\eta_{-\theta}, \eta_{+\theta}].$$

Therefore, letting $p_j = (\gamma_j)^{n_R}$, we have $c_j \sim \text{Ber}(p_j)$, where Ber denotes the Bernoulli distribution. Also, recall that every number in $Y_{R,j}$ is chosen uniformly at random from $[0, 1]$ and therefore these Bernoulli distributions are independent from each other. Therefore, we can apply Lemma 6 below to conclude that $C_{K_a} \approx \epsilon, \delta C_{K_a} + s$.

**DP of Additive Poisson Binomial distribution.** For $j \in [n]$, consider $c_j \sim \text{Ber}(p_j)$. With $p_j = \{p_j\}_{j=1}^n$, let $\text{PB}(n, p_j)$ denote the distribution of $\sum_{j \in [n]} c_j$. This distribution is called a Additive Poisson Binomial distribution.

**Lemma 6.** Consider an Additive Poisson Binomial distribution $\text{PB}(n, p_j)$, where $n \in \Omega(\kappa)$ and for each $p_j$, it holds that $p_j \in [1/2 - \theta, 1/2 + \theta]$ where $\theta \in (0, 1/2)$ is a constant independent of $\kappa$. Then, for any constant $\epsilon$ and $s \leq \lg \lg \kappa$, there are $a, b \in [n]$ such that

- For any $\ell \in [a, b]$, $e^{-\epsilon} \leq \frac{\Pr[\text{PB}(n, p_j) = \ell]}{\Pr[\text{PB}(n, p_j) = \ell]} \leq e^\epsilon$.
- For any $\ell \not\in [a + s, b]$, $\Pr[\text{PB}(n, p_j) = \ell] = \text{negl}(\kappa)$ and $\Pr[\text{PB}(n, p_j) + s = \ell] = \text{negl}(\kappa)$.

## 7 Strong Chain rule

Our proof of Theorem 3 requires a special case of the following theorem, corresponding to a strong chain rule for specific leakage functions $\ell_1(\cdot), \ldots, \ell_n(\cdot)$. By formalizing the properties needed from $\ell_1(\cdot), \ldots, \ell_n(\cdot)$ for the proof of the special case to go through, we are able to arrive at the generalization presented in this section. We state the generalized version of the theorem since we believe it may find future applications in leakage-resilient cryptography.

Recall that we consider a block-by-block random variable $R = (R_1, \ldots, R_n)$, and (potentially randomized) leakage functions $\ell_1(\cdot), \ldots, \ell_n(\cdot)$ with randomness $\rho_1, \ldots, \rho_n$. You can think of the blocks as coming in a streaming fashion in order of $R_1, R_2, \ldots, R_n$.

Loosely speaking, the properties we require of the leakage functions are that the $i$-th leakage $\ell_i$ can be computed given $R_i, \rho_j$ as well as all the outputs of $(\ell_{i+1}, \ell_{i+2}, \ldots, \ell_n)$ and that the total number of valid sequences of leakages from $\ell_1(\cdot), \ldots, \ell_n(\cdot)$ is sufficiently small (see Property 1 in Theorem 4).

Our theorem below states the existence of a spoiling function $f(\cdot)$ with certain properties, as well as properties of the random variables $(R_1, \ldots, R_n)$ and $(\rho_1, \ldots, \rho_n)$ conditioned on the output of the spoiling function $f(R)$.

The properties of $(R_1, \ldots, R_n)$ and $(\rho_1, \ldots, \rho_n)$ are roughly the following: (1) There exist disjoint sets $V, W$ such that $V \cup W = [n]$ that are determined by $f(R)$. (2) Blocks $\{R_i\}_{i \in V}$ have high min-entropy conditioned on $f(R)$. (3) Blocks
\{R_i\}_{i \in W}$ have small support size (low max-entropy) conditioned on $f(R)$. (4) For $i \in V$, the random strings $\rho_i$ are uniform random and independent conditioned on $f(R)$. (See Properties (5)-(8) in Theorem 4).

The properties of $f(\cdot)$ are roughly the following: (1) The failure probability (outputting $\bot$) is small. (2) As long as the total number of valid sequences of leakages from $\ell_1(\cdot), \ldots, \ell_n(\cdot)$ is sufficiently small, the image size of $f$ is small. This property ensures that we do not lose too much of the total min-entropy of $R$ by releasing $f(R)$ (3) The leakages $\{\ell_i(\cdot)\}_{i \in W}$ can be computed given $f(R)$. (See Properties (2)-(4) in Theorem 4).

The main difference between our spoiling lemma and prior ones is that our min and max entropy guarantees on $R = (R_1, \ldots, R_n) \mid f(R)$ hold even with respect to additional leakage $\{\ell_i\}_{i \in W}$ which is included in the spoiled bits $f(R)$.

**Theorem 4 (Block structures with few bits spoiled and leakage).** Let $U = U_1 \times \cdots \times U_n$ be a fixed universe and $R = (R_1, \ldots, R_n)$ be a sequence of (possibly correlated) random variables where each $R_i$ is over $U_i$ (and all are disjoint) and $|U_i| = \ell$ for all $i$. Let $\rho_1, \ldots, \rho_n$ be a sequence of uniformly random strings over $\{0, 1\}^m$ and let $\ell_1(\cdot), \ldots, \ell_n(\cdot)$ be leakage functions. Then, for any $\epsilon \in (0, 1)$, any $\delta > 0$ and any $c \in [2^b, \ell/2^b]$, there exists a spoiling leakage function $f(R)$ that satisfies the following properties.

1. A sequence $\beta_1, \ldots, \beta_n$ is valid if for all $i \in V$, $\beta_i = \bot$ and for all $i \in W$, $\beta_i = \ell_i(R_i, \rho_i, \beta_{>i})$, where $\beta_{>i} = (\beta_{i+1}, \ldots, \beta_n)$. We require that the number of valid sequences $\beta_1, \ldots, \beta_n$ is at most $B$.
2. It holds that $\Pr[f(R) = \bot] \leq c\ell$.
3. $|\text{Im}(f)| \leq B \cdot (2(\log(\ell) + \log(1/\epsilon))/\delta)^n$.
4. Conditioned on any $y \in \text{Im}(f) \setminus \{\bot\}$, for all $i \in W$, the leakage $\ell_i(R_i, \rho_i, \beta_{>i})$ can be computed from $y$. Here, $\beta_j = \bot$ if $j \in V$ and $\beta_j = \ell_j(R_j, \rho_j, \beta_{>j})$ otherwise.
5. Let $\text{Im}(f)$ be the set of images of $f$. Every $y \in \text{Im}(f) \setminus \{\bot\}$ specifies two disjoint sets $V$ and $W$ such that $V \cup W = [n]$.
6. Conditioned on any $y \in \text{Im}(f) \setminus \{\bot\}$, for every $i \in V$, every element in distribution $R_i \mid R_{<i}$ has low probability weight, i.e.,

\[
\forall y \in \text{Im}(f) \setminus \{\bot\}, \forall r \text{ s.t. } f(r) = y, \forall i \in V : \Pr[R_i = r_i \mid R_{<i} = r_{<i}, y] \leq \frac{2^b}{c}.
\]

7. Conditioned on any $y \in \text{Im}(f) \setminus \{\bot\}$, for every $i \in W$, it holds that $R_i \mid R_{<i}$ has small support size, i.e.,

\[
\forall y \in \text{Im}(f) \setminus \{\bot\}, \forall r \text{ s.t. } f(r) = y, \forall i \in W : \Pr(|r_i : \Pr[R_i = r_i \mid R_{<i} = r_{<i}, y] \geq 0|) \leq 2^b \cdot c.
\]

8. $\{\rho_i\}_{i \in V}$ are distributed independently and uniformly at random conditioned on $f(R)$. 

21
Typically, one would like to set $c$ as large as possible, while ensuring that the size of $V$ remains above some threshold. The achievable tradeoffs between the setting of $c$ and the size of $V$ are determined by the min-entropy of $R$ before the spoiling bits $f(R)$ are released. For our applications, we require $c = n^{1.5}$ and $|V| \geq n/3$. We show that our min-entropy assumption on $R$ implies that this parameter setting is achievable in Section F.3.

8 Highlights of Proof of Theorem 3

Due to the lack of space, we highlight only the important parts of the proof of Theorem 3. The full proof can be found in Appendix.

Remember that the adversary holds set $A$ of size $n_A$. The honest party holds set $B$ of size $n_B$. The intersection $I := A \cap B$ has size $n_I$. The secret set held by the honest party $R := B \setminus I$ has size $n_R$. We set $n'_R := n_R/3$; looking forward, it is the size of a subset $R' \subset R$, each of whose elements has high remaining min-entropy even after leakage (that we will define in the proof) is considered.

8.1 Min-hash Graph

Consider running the min-hash protocol $\pi_{PH}$ with $k$ iterations such that $k_g$ of them belong to $G_\emptyset$. For this, we consider all the hash outputs in two different stages and define the following sets:

$$H_1 = \{h_j(A+x^*)\}_{j=1}^k, \quad H_2 = \{h_j(U \setminus A+x^*)\}_{j=1}^k.$$

Since we are in the random oracle model, each hash value is chosen uniformly at random. For our analysis, we construct the following bipartite graph $(X, Y, E)$, which we call the min-hash graph, based on the sets $A, I$ and $x^*$ along with the hash functions as follows:

MinhashG$_{H_1}(A, I, x^*, H_2)$:
1. Set $X = U \setminus A+x^*$. In other words, the graph considers all potential elements that could be in $B$. A distribution of $B$ is equivalent to a distribution of how to choose $n_B$ nodes in $X$.
2. Use $H_1$ to determine $G_\emptyset$ and set $Y = G_\emptyset$. In other words, $Y$ contains all good iterations that could potentially increase the final count.
3. Let $p_j = \min h_j(I)$. Use $H_2$ to determine the set of edges:

$$E = \{(i,j) : (i,j) \in X \times Y \text{ and } h_j(x_i) < p_j\}.$$

In other words, existence of an edge $(i,j)$ means that if node $i$ belongs to input $B$, iteration $j$ will not contribute to the final count.
4. Output the resulting bipartite graph $(X, Y, E)$. 

22
Fig. 5: Min-hash graph

Table 1: Example Hash Functions

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>h_1</td>
<td>0.83</td>
<td>0.25</td>
<td>0.77</td>
<td>0.85</td>
<td>0.93</td>
<td>0.35</td>
<td>0.86</td>
<td>0.92</td>
<td>0.49</td>
<td>0.21</td>
<td>0.5</td>
</tr>
<tr>
<td>h_2</td>
<td>0.62</td>
<td>0.83</td>
<td>0.27</td>
<td>0.59</td>
<td>0.63</td>
<td>0.26</td>
<td>0.4</td>
<td>0.26</td>
<td>0.72</td>
<td>0.36</td>
<td>0.6</td>
</tr>
<tr>
<td>h_3</td>
<td>0.68</td>
<td>0.11</td>
<td>0.67</td>
<td>0.29</td>
<td>0.82</td>
<td>0.3</td>
<td>0.62</td>
<td>0.23</td>
<td>0.67</td>
<td>0.35</td>
<td>0.7</td>
</tr>
<tr>
<td>h_4</td>
<td>0.02</td>
<td>0.43</td>
<td>0.22</td>
<td>0.58</td>
<td>0.69</td>
<td>0.67</td>
<td>0.93</td>
<td>0.56</td>
<td>0.11</td>
<td>0.42</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Example. Let the universe be \( \mathcal{U} = [11] \). Let \( A = \{1, 2, 3, 4\} \), \( I = \{2, 3\} \), \( x^* = 11 \). Let the threshold range for the \( \theta \)-good iterations be \([0.2, 0.7]\). Assume that our protocol runs in 4 iterations using the hash functions defined in Table 1.

Figure 5 shows the constructed min-hash graph. In particular, we have \( X = \{5, 6, \ldots, 10\} \). We have \( Y = \{h_1, h_2\} \); \( h_3 \) has been ruled out since \( p_3 = h_3(2) = 0.11 \notin [0.2, 0.7] \), and \( h_4 \) has been ruled out because \( \min h_4(A) \neq \min h_4(I) \). Moreover, we have \( p_1 = h_1(2) = 0.25 \) and \( p_2 = h_2(3) = 0.27 \). Note that \((8, h_2) \in \mathcal{E} \), because \( h_2(8) < p_2 \).

8.2 Fixed Subsets of Secret Items and Good Iterations

We use the min-hash graph and analyze DDP of \( \pi_{\text{PH}} \) for subsets of secret items and good iterations. We first fix items and consider a noise distribution over the choice of hash functions. Extending this, in the next section, we will consider the noise over a distribution of items.

Edges in a min-hash graph. In the random oracle model, each answer in \( H_2 \) is chosen uniformly at random. Therefore, given \( H_1 \), the probability (over the choice of \( H_2 \)) that an edge \((i, j)\) forms is exactly equal to \( p_j \). Moreover, the probability that \((i, j)\) forms is independent of the probability that any other edge in the graph forms.

Distribution of min-hash graphs over the choice of \( H_2 \). Fix the hash answers \( H_1 \) to fix \( \mathcal{X} \) and \( \mathcal{Y} \). Fix any positive integer \( \hat{n} \leq n_R \) and the set \( T \subseteq \mathcal{Y} \) of iterations with \(|T| = \hat{k} \leq k_g \). For a fixed set \( \hat{R} \subseteq \mathcal{X} \) of \( \hat{n} \) nodes on the left, let \( \mathcal{G}_Y^{\hat{R}} \) be the set of all min-hash graphs that cover exactly the set \( Y \subseteq T \). More formally, given a set of edges \( \mathcal{E} \), define the set of destination nodes as \( \text{dest}(\mathcal{E}) = \{j : (i, j) \in \mathcal{E}\} \) respectively. Define the set of all possible min-hash subgraphs in which the set of destination nodes is exactly \( Y \) (from \( \hat{R} \)).

\[
\mathcal{G}_Y^{\hat{R}} = \{(\hat{R}, Y, \mathcal{E}') : \mathcal{E}' \in \hat{R} \times Y, \text{dest}(\mathcal{E}') = Y\}.
\]

In this section, we are interested in the probability over the choice of the hash functions that the final count is reduced by exactly \( r \) due to the elements of a fixed set \( \hat{R} \) over the \( \hat{k} \) iterations in \( T \). In particular, we define

\[
E_{T, r}^{\hat{R}} = \sum_{Y \subseteq T, |Y| = r} \sum_{G \in \mathcal{G}_Y^{\hat{R}}} p(G),
\]

where \( p(G) \) is defined as

\[
p(G) = \Pr_{H_2} [G_0 \leftarrow \text{MinhashG}_{H_1}(A, I, x^*, H_2); G \text{ is a subgraph of } G_0].
\]
As explained above, in the random oracle model, the probability depends only on the size of the sets \( \hat{R} \) and \( T \) (i.e., not the actual identity of the set). Therefore, we will often use the notation \( E_{\hat{R},r}^{\hat{n}} = E_{\hat{T},r}^{\hat{n}} \) when \( |\hat{R}| = \hat{n} \) and \( |T| = \hat{k} \).

Observe that \( E_{\hat{n},\hat{k},r}^{\hat{n}} \) is another way of representing an Additive Poisson Binomial distribution. That is, \( E_{\hat{n},\hat{k},r}^{\hat{n}} = \Pr\{\text{PB}(\hat{k}, p_J) = r\} \).

Therefore, based on Lemma 6, we have the following:

**Corollary 1.** Fix \( H_1 \). Let \( s = \lg \lg \kappa \). For any constant \( \epsilon \), and any \( \hat{k} \in \Omega(\kappa) \), there are \( a, b \in [\hat{k}] \) such that

- For any \( r \not\in [a + s, b] \), then \( E_{\hat{n},\hat{k},r}^{\hat{n}} \) and \( E_{\hat{n},\hat{k},r-s}^{\hat{n}} \) are both negligible in \( \kappa \).

- For any \( r \in [a, b] \), then it holds \( e^{-\epsilon/3} \leq \frac{E_{\hat{n},\hat{k},r}^{\hat{n}}}{E_{\hat{n},\hat{k},r-s}^{\hat{n}}} \leq e^{\epsilon/3} \).

### 8.3 DDP over Distribution with Geometric Collision

**Additional notation.** In our analysis, we focus on only \( 1/3 \)-fraction of high min-entropy elements of the secret set \( R \) to deal with the leakage scenario, and consider the Geometric Collision Property only for this subset of elements, which we denote by \( R' \).

Consider any min-hash graph \( G = (X, Y, E) \). For any set \( T \) of iterations of size \( \hat{k} \) and any integer \( r \), let \( I_{R',T,r} \) be the indicator random variable that is set to 1 if set \( R' \) achieves total noise \( r \) among the iterations in \( T \).

In this section, we consider a distribution \( \bar{D} \) of the secret set \( R' \). We define the following measure

\[
D_{T,r}(\bar{D}) := \Pr_{R \sim \bar{D}}[I_{R',T,r}] = \sum_{R'} \Pr_{R \sim \bar{D}}[R'] \cdot I_{R',T,r}.
\]

**Geometric collision property.** Observe that \( D_{T,r} \) corresponds to the probability (over \( D \)) that \( R' \) contributes to the noise pattern \( r \). We would like to show the following:

For any fixed \( H_2 \) and over distribution \( D \), it holds that \( D_{T,r} \) and \( D_{T,r-1} \) (and ultimately \( D_{T,r-s} \)) are close, except with the tail case of \( r \) whose probability weight is negligible.

The universal quantifier for \( H_2 \) in the above can be slightly relaxed so that the condition holds with all but small probability over the choice of \( H_2 \). We observe that the above condition can be captured by showing that \( D_{T,r} \) is close to its mean \( E_{\hat{n},\hat{k},r}^{\hat{n}} \) (and then taking advantage of the property of \( E_{\hat{n},\hat{k},r}^{\hat{n}} \) described in Corollary 1). This is essentially to show that \( D_{T,r} \) is concentrated around its mean. We could try to apply Chernoff bound to show the concentration property,
but we cannot because \( I_{R_i', T, r} \) and \( I_{R_j', T, r} \) are not necessarily independent if \( R_i' \cap R_j' \neq \emptyset \). Therefore, we instead use Chebyshev for bounding the tail, which requires \( D_{T, r} \) to have small variance. Towards this goal, we introduce a notion called geometric collision property.

**Definition 7 (Geometric Collision Property).** Let \( \bar{D} \) be a distribution over sets of size \( n'R \). We say that \( \bar{D} \) has the Geometric Collision Property if for all \( z \in [n'R] \)

\[
\Pr_{R_i', R_j' \sim \bar{D}}[|R_i' \cap R_j'| = z] \leq \left( \frac{1}{\sqrt{n'R}} \right)^z.
\]

Based on this property, we show the following lemma.

**Lemma 7.** Let \( D \) be a distribution over sets of size \( n'R \) with geometric collision property. For any set \( T \) of size \( \hat{k} \in \Omega(\kappa) \), there exist \( a, b \in [\hat{k}] \), such that with probability \( 1 - O(\hat{k} \log^{3} \kappa) \sqrt{n'R} \) over choice of \( H_2 \), the following holds:

- For all \( r \not\in [a + s, b] \), \( D_{T, r} \) is negligible, where \( s = \log \log \kappa \).
- For all \( r \in [a, b] \), \( e^{-\epsilon/3} E_{\hat{k}, r}^{n'R} \leq D_{T, r} \leq e^{\epsilon/3} E_{\hat{k}, r}^{n'R} \).

### 8.4 Distribution with the Geometric Collision Property

Recall that the receiver’s input is chosen from the following distribution:

- For each \( i \in (n_I, n_B] \), choose \( x_i^B \) uniformly at random from the universe \( U_i \).

We assume that for \( i \neq j \), \( U_i \) and \( U_j \) are disjoint with the same cardinality \( |U_i| = |U_j| := \ell \).

Indeed, the above distribution has the geometric collision property as long as \( \ell \geq n_R \cdot \sqrt{n_R} \) (recall \( n_R = n_B - n_I \)). When considering two random sets \( R_0' \) and \( R_i' \) of size \( n'R' = n'R/3 \) where their elements are from the above distribution, each position \( i \) will have collision with probability \( 1/\ell \), so we have

\[
\Pr[|R_0' \cap R_i'| = z] = \left( \frac{n_R'}{z} \right) \left( \frac{1}{\ell} \right)^z \cdot \left( \frac{\ell - 1}{\ell} \right)^{n_R' - z} \leq \left( \frac{e \cdot n_R'}{z \ell} \right)^z \leq \left( \frac{1}{\sqrt{n_R}} \right)^z .
\]

The main issue is that we need to deal with the leakage stemming from the fact that the hash functions are public. Therefore, we need to consider a more general class of distributions that captures the leaked version of the above distribution and show that they still possess the geometric collision property.

For brevity, we omit the subscript \( R \) and denote \( n = n_R \) and \( n' = n'R \).

Applying the strong chain rule, we can show that the aforementioned distribution still contains \( n' \) blocks each of which maintains high min-entropy, even after the leakage is considered.

25
Lemma 8. We consider a min-hash graph $G = (\mathcal{X}, \mathcal{Y}, \mathcal{E})$ constructed from Minhash $H_{H_1(A, I, x^*, H_2)}$, while focusing on a single bundle of good iterations. Let $\mathcal{U} = U_1 \times \cdots \times U_n$ be a fixed universe and $R = (R_1, \ldots, R_n)$ be a sequence of (possibly correlated) random variables where each $R_i$ is over $U_i$ (and all are disjoint) and $|U_i| = \ell$ for all $i$. Let $\mathcal{D}_{\text{leak}}$ be a distribution over $R$ that has min-entropy at least $\frac{8n}{9} \lg(\ell) + n$. Then, there is a spoiling leakage function $f'_{h_G}$ such that with all but negligible probability over $R \sim \mathcal{D}_{\text{leak}}$, the distribution $\tilde{\mathcal{D}} := \mathcal{D}_{\text{leak}} | f'_{h_G}(R)$ has the following property:

There exists a set $\{i_1, \ldots, i_n\}$ of size $n'$ such that for $v \in [n']$, and any $(r_{i_1}, \ldots, r_{i_{v-1}})$ in the support of $\mathcal{D}([i_v - 1])$, the random variable $R_{i_v} \sim \mathcal{D}([i_v]) | R_{i_1} = r_{i_1}, \ldots, R_{i_{v-1}} = r_{i_{v-1}}$ has min-entropy at least $1.5 \lg(n)$.

Given the properties of $\tilde{\mathcal{D}}$ stated in Lemma 8 by taking advantage of the $n'$ high min-entropy blocks, we show that $\tilde{\mathcal{D}}$ has the Geometric Collision Property.

Lemma 9. Assume $\tilde{\mathcal{D}}$ has the following properties (from the conclusion of Lemma 8):

- There exists a set $\{i_1, \ldots, i_n\}$ of size $n'$ such that for $v \in [n']$, and any $(r_{i_1}, \ldots, r_{i_{v-1}})$ in the support of $\tilde{\mathcal{D}}$ (again we abuse notation and consider $\tilde{\mathcal{D}}$ to be a distribution over streams $R' = R_{i_1}, \ldots, R_{i_{n'}}$), the random variable $R_{i_v} \sim \tilde{\mathcal{D}} | R_{i_1} = r_{i_1}, \ldots, R_{i_{v-1}} = r_{i_{v-1}}$ has min-entropy at least $1.5 \lg(n)$.

Then $\tilde{\mathcal{D}}$ (viewed as a distribution over sets) has the Geometric Collision Property (see Definition 7).

9 Empirical Evaluation

We conduct empirical evaluations of our protocols to determine the proper parameter ranges for privacy. Fig. 6 shows the trade-off between $\epsilon$ and $\delta$ with respect to different intersection sizes $n_I$ and the number of iterations when $n_A = n_B = 10^6$. In Fig. 7 we show the number of iterations required to achieve DP or DDP for a given Jaccard index for various values of $\epsilon, \delta$ both in the curator setting and in the two-party setting where non-intersecting items have high min-entropy. Note that we achieve the best privacy claim when the Jaccard index is around 0.5, where the matching of the hash on two sets is most “uncertain”. In Fig. 8 we show how the expected error of minhash changes with respect to the Jaccard index and number of iterations. Note that regardless of the setting, the output of minhash follows a binomial distribution, with randomness taken over the choices of hash functions. More specifically, the expected value of this binomial distribution equals the Jaccard index. Therefore, the expected error of minhash is exactly the standard deviation of the binomial distribution.
Fig. 6: \( n_A = n_B = 10^6 \). Left: Curator setting. Right: Two-party setting with high min-entropy

Fig. 7: \( n_A = n_B = 10^6 \). Left: Curator setting. Right: Two-party setting with high min-entropy

Fig. 8: Expected error of minhash. Left: Trend with various numbers of iterations. (We only plot curves with a Jaccard index up to 0.5, as the expected error is symmetric around 0.5.) Right: Trend with various Jaccard Index.
References


A  Hockey stick divergence

We first review hockey stick divergence [38], also known as elementary divergences [49] or $\alpha$-distance [3]. In this paper, we only consider discrete sample space, although many of arguments naturally extend to the continuous space.

**Definition 8.** The hockey-stick divergence between two probability measures $P, Q$ over $\mathbb{Z}$ is defined as:

$$D_{\alpha}^{hs}(X, Y) = \sup_{S \subseteq \mathbb{Z}} (X(S) - \alpha Y(S)) = \sum_{z \in \mathbb{Z}} [(X(z) - \alpha Y(z)]_+,$$

where $\alpha \geq 1$ and $[x]_+ = \max\{x, 0\}$.

We observe that the following holds directly from the definition of the hockey-stack divergence.

**Corollary 2.** For any probability measures $X, Y$ over $\mathbb{Z}$ and for any $\epsilon, \delta$, it holds

$$X \approx_{\epsilon, \delta} Y \text{ if and only if } D_{\epsilon}^{hs}(X, Y) \leq \delta \text{ and } D_{\epsilon}^{hs}(Y, X) \leq \delta.$$

Therefore, the hockey-stick divergence captures the inequalities between output distributions from neighboring inputs. We use hockey-stick divergence to analyze the privacy loss.

**The hockey-stick divergence as a $f$-divergence.** It is known that the hockey-stick divergence is a kind of $f$-divergence [3]. Therefore, the hockey-stick divergence satisfies the joint convexity and data processing inequality [312].

**Lemma 10 (Joint convexity).** For all $0 \leq \lambda \leq 1$ and $\alpha \geq 1$, any probability measures $X_1, X_2, Y_1, Y_2$ satisfy

$$D_{\alpha}^{hs}(\lambda X_1 + (1 - \lambda) X_2, \lambda Y_1 + (1 - \lambda) Y_2) \leq \lambda D_{\alpha}^{hs}(X_1, Y_1) + (1 - \lambda) D_{\alpha}^{hs}(X_2, Y_2).$$

The data processing inequality property guarantees that post-processing cannot increase the divergence.

**Lemma 11 (Data processing inequality.).** For any probability measures $X, Y$ over $\mathbb{Z}$ and any function $g$ with domain $\mathbb{Z}$, we have

$$D_{\alpha}^{hs}(g(X), g(Y)) \leq D_{\alpha}^{hs}(X, Y).$$
We set \(a = \frac{np + s(1-p)e^{\epsilon/s}}{e^{\epsilon/s} - (1-p)p} \); we chose \(a\) so that \(\frac{n-a}{a-s} = e^{\epsilon/s} \cdot \frac{1-p}{p}\) in order to work out the following:

\[
\begin{align*}
\Pr_{B(n,p)}[a] &= \binom{n}{a} p^a (1-p)^{n-a} \\
&= \frac{(a-s)!}{a!} \cdot \frac{(n-a)!}{(n-a)!} \cdot \left(\frac{p}{1-p}\right)^s
\end{align*}
\]

\[
\Pr_{B(n,p) + s}[a] = e^\epsilon.
\]

We also set \(b = \frac{e^{\epsilon/s} np}{(1-p)+e^{\epsilon/s}p}\); we chose \(b\) so that \(\frac{n-b}{b-s} = e^{-\epsilon/s} \cdot \frac{1-p}{p}\) in order to work out the following:

\[
\begin{align*}
\Pr_{B(n,p)}[b] &= \binom{n}{b} p^b (1-p)^{n-b} \\
&= \frac{(b-s)!}{b!} \cdot \frac{(n-b+s)!}{(n-b)!} \cdot \left(\frac{p}{1-p}\right)^s
\end{align*}
\]

\[
\Pr_{B(n,p) + s}[b] = e^{-\epsilon}.
\]

To show the second requirement, it suffices to show that \(\Pr_{B(n,p)}[X \leq a + s] = \text{negl}(\kappa)\); the case \(\Pr_{B(n,p)}[X \geq b] = \text{negl}(\kappa)\) holds similarly.

Let \(\mu := np \in \Theta(\kappa)\) and let \(d = 1 - (a + s)/\mu\). By applying the Chernoff bound, we have

\[
\Pr_{X \leftarrow B(n,p)}[X \leq a + s] = \Pr[X \leq (1 - d)\mu] \leq \exp(-d^2 \mu/2).
\]
To see the asymptotic measure of \( d \), let \( t = s(1 - p) \cdot e^{\epsilon/s} \) and then we have

\[
a = \frac{\mu + t}{e^{\epsilon/s}(1 - p) + p}.
\]

Then, we have

\[
d = \frac{\mu - a}{\mu} - \frac{s}{\mu} = \frac{\mu((1 + \epsilon/s) \cdot (1 - p) + p) - \mu - t}{\mu \cdot e^{\epsilon/s}} - \frac{s}{\mu}
\]

\[
\geq \frac{(\epsilon/s) \cdot (1 - p)}{e^{\epsilon/s}} - t + \frac{s}{\mu}
\]

\[
= \Theta(1/\lg \lg \kappa - \tilde{O}(1/\kappa))
\]

Since we have \( d = \Omega(1/\lg \lg \kappa) \) and \( \mu \in \Theta(1/\kappa) \), \( \Pr_{X \leftarrow B(n,p)}[X \leq a + s] \) is negligible in \( \kappa \).

### C Proof of Lemma 6

Let \( \eta - \theta = 1/2 - \theta \) and \( \eta + \theta = 1/2 + \theta \). For brevity, we let \( C \) denote \( PB(n, p_j) \). For any distribution \( D \), let \( P_D \) denote the probability measure with respect to \( D \). We first show that for any \( \epsilon > 0 \), it holds

\[
D^{hs}_{e^\epsilon}(P_C, P_{C+1}) \leq \max_{j \in [n]} \left( \min \left( P_B([\frac{n}{2}, \eta + \theta]), P_B([\frac{n}{2}, \eta - \theta]) + 1 \right), \min \left( P_B([\frac{n}{2}, \eta - \theta]), P_B([\frac{n}{2}, \eta + \theta]) + 1 \right) \right).
\]

To show the above, we follow a similar structure to the proof of [11 Theorem 3.3], which analyzes the Renyi divergence of a slightly different version of (non-additive) Poisson binomial mechanism. We also rely on the joint convexity\(^5\) (Lemma 10) and data processing inequality (Lemma 11) to upper bound the hockey-stick divergence of two Poisson binomial distributions with that of two binomial distributions.

We start with showing that the upper bound of the hockey-stick divergence is reached at extreme points. The proof is similar to [11 Lemma 3.5].

**Lemma 12.**

\[
D^{hs}_{e^\epsilon}(P_C, P_{C+1}) \leq \max_{j \in [n]} D^{hs}_{e^\epsilon}(P_{B(j, \eta - \theta) + B(n-j, \eta + \theta)}, P_{B(j, \eta - \theta) + B(n-j, \eta + \theta) + 1}),
\]

\[
\text{\textsuperscript{5}} [11] \text{ actually uses joint quasi-convexity, which is implied by joint convexity but not vice versa.}
\]

34
We have
\[ P_C = \lambda P_{C_n + \text{Ber}(\eta_{-\theta})} + (1 - \lambda) P_{C_n + \text{Ber}(\eta_{+\theta})}. \]

Now using Lemma 10, we have
\[
D_{c^t}^{\text{hs}}(P_{C}, P_{C+1}) 
\leq \lambda D_{c^t}^{\text{hs}}(P_{C_n + \text{Ber}(\eta_{-\theta})}, P_{C_n + \text{Ber}(\eta_{-\theta})+1}) + (1 - \lambda) D_{c^t}^{\text{hs}}(P_{C_n + \text{Ber}(\eta_{+\theta})}, P_{C_n + \text{Ber}(\eta_{+\theta})+1})
\leq \max \left\{ D_{c^t}^{\text{hs}}(P_{C_n + \text{Ber}(\eta_{-\theta})}, P_{C_n + \text{Ber}(\eta_{-\theta})+1}), D_{c^t}^{\text{hs}}(P_{C_n + \text{Ber}(\eta_{+\theta})}, P_{C_n + \text{Ber}(\eta_{+\theta})+1}) \right\}
\]

Repetitively applying the joint convexity on the remaining \( n - 1 \) random variables yields the following
\[
D_{c^t}^{\text{hs}}(P_Y, P_{Y+1}) \leq \max_{j \in [n]} D_{c^t}^{\text{hs}}(P_{N_j}, P_{N_j+1}),
\]
where \( N_j \sim \text{B}(j, \eta_{-\theta}) + \text{B}(n - j, \eta_{+\theta}). \)

Next, we apply data processing inequality to simplify (3) from above lemma.

**Lemma 13.** \( j \) is upper bounded by
\[
\max \left\{ D_{c^t}^{\text{hs}}(P_{\text{B}([\frac{n}{2}], \eta_{-\theta})}, P_{\text{B}([\frac{n}{2}], \eta_{-\theta})+1}), D_{c^t}^{\text{hs}}(P_{\text{B}([\frac{n}{2}], \eta_{-\theta})}, P_{\text{B}([\frac{n}{2}], \eta_{-\theta})+1}) \right\}
\]

**Proof.** The proof is similar to [11, Lemma 3.6]. We apply the data processing inequality. More specifically, consider two distributions \( X, Y \) and then adding a binomial random variable \( Z \) as the post-processing step. Then, the addition of \( Z \) doesn’t increase the divergence. In other words,
\[
D_{c^t}^{\text{hs}}(P_X, P_Y) \geq D_{c^t}^{\text{hs}}(P_{X+Z}, P_{Y+Z}).
\]

Now, consider (3) and let \( j^* \) be the index that leads to the maximum.

- If \( j^* \leq n/2 \), we have
\[
D_{c^t}^{\text{hs}}(P_{\text{B}([\frac{n}{2}], \eta_{+\theta})}, P_{\text{B}([\frac{n}{2}], \eta_{+\theta})+1}) 
\geq D_{c^t}^{\text{hs}}(P_{\text{B}(j^*, \eta_{-\theta})+\text{B}(n-j^*, \eta_{+\theta})}, P_{\text{B}(j^*, \eta_{-\theta})+\text{B}(n-j^*, \eta_{+\theta})+1}).
\]

This is because \( n/2 \leq n - j^* \). In this case, \( Z = \text{B}(j^*, \eta_{-\theta}) + \text{B}(n - j^* - \left\lfloor \frac{n}{2} \right\rfloor, \eta_{+\theta}). \)
Likewise, if $j^* \geq n/2$, we have
\[
D_e^{bs}\left(P_B([\frac{n}{2}], \eta - \delta), P_B([\frac{n}{2}], \eta + \delta + 1)\right)
\geq D_e^{bs}\left(P_B(j^*, \eta - \delta) + B(n - j^*, \eta + \delta); P_B(j^*, \eta - \delta) + B(n - j^*, \eta + \delta + 1)\right).
\]

We extend the above to upper bound the hockey-stick divergence between
probability measures differed by an integer amount greater than 1, i.e., $P_C$ and $P_{C+s}$ for $s > 1$.

**Corollary 3.** For any $\epsilon > 0$, we have
\[
D_e^{bs}(P_C, P_{C+s}) \leq \max\left(D_e^{bs}\left(P_B([\frac{n}{2}], \eta - \delta), P_B([\frac{n}{2}], \eta + \delta + s)\right), D_e^{bs}\left(P_B([\frac{n}{2}], \eta - \delta), P_B([\frac{n}{2}], \eta + \delta)\right)\right).
\]

Finally, to give a bound on the divergence, we can apply Lemma 3 to argue
binomial distribution hides the small sensitivity. Specifically, as $[\frac{n}{2}] \in \Theta(\kappa)$ and $s = \lg \lg \kappa$, we can claim $(\epsilon, \delta)$-DDP with $\delta = \text{negl}(\kappa)$.
Similarly, it holds that $D_e^{bs}(P_{C+s}, P_C) \leq \text{negl}(\kappa)$.

\[\square\]

**D Proof of Theorem 3**

**On the definition of a \(\theta\)-good iteration.** We keep the same definition of a \(\theta\)-good iteration, except we set the exponent to $1/n_R'$, instead of $1/n_R$, where $n_A, n_B, n_I, n_R, n_R'$ are defined above. In particular,

- $\min h(A) = \min h(I)$.
- $\min h(I) \in \left[1 - (\frac{1}{2} + \theta)^t, 1 - (\frac{1}{2} - \theta)^t\right]$, where $t = \frac{1}{n_R'}$.

Further, we require $\theta \leq 1/10$.

**Bundle of good iterations \(K_\theta\).** The total number of iterations in the min-hash protocol $\mathcal{PH}$ is $k = \Omega(\kappa \cdot \lg \lg \kappa)$. We require that $n_R/k^2 = \Omega(\kappa)$.

Using Lemma 4 with all but negligible probability, at least $\Omega(\kappa \cdot \lg \lg \kappa)$ iterations are \(\theta\)-good. Recall that $G_\theta$ denotes the set of \(\theta\)-good iterations, and $K_\theta = G_\theta \setminus S_x\star$. We set $k_\theta = |K_\theta|$. Of these $k_\theta$ number of \(\theta\)-good iterations (except in $S_x\star$), we further divide them into $u = \lg \lg \kappa$ bundles, each of which is of size $k_\theta = \Omega(\kappa)$. Those bundles are denoted by $K_{\theta,1}, \ldots, K_{\theta,u}$. We also let $K_{bad} := K_{\theta} \cup \bigcup S_x\star$.

**Random variables for the protocol output.** Let $out_{bad}^+$ be the protocol’s match count for sets $A, B_{+x\star}$ w.r.t. the hash functions in $K_{bad}$:
\[
out_{bad}^+ := |\{j \in K_{bad} : \min h_j(A) = \min h_j(B_{+x\star})\}|.
\]

Likewise, let $out_{bad}$ be the number of matches for sets $A$ and $B$ (instead of $B_{+x\star}$) in iterations in $K_{bad}$. Similarly, for $i \in [u]$, we let $out_{i}^+$ and $out_{i}$ denote the output
for the $i$-th bundle, with or without $x^*$ respectively. Note that $\text{out}_i^+ = \text{out}_i$, since we ruled out $S_x^*$ from $K_\theta$. Note that the final output of the min-hash protocol for input $B_{+x^*}$ is equal to $\text{out}_\text{bad}^+ + \sum_{i=1}^u \text{out}_i$; the final output for input $B$ is $\text{out}_\text{bad} + \sum_{i=1}^u \text{out}_i$. Let

$$\text{out} = \text{out}_\text{bad}^+ \| \text{out}_\text{bad} \| \text{out}_1 \| \cdots \| \text{out}_u.$$  

We also consider the output with the $i$-th bundle missing; that is, for $i \in [u]$ let

$$\text{out}_{-i} = \text{out}_\text{bad}^+ \| \text{out}_\text{bad} \| \text{out}_1 \| \cdots \| \text{out}_{i-1} \| \text{out}_{i+1} \| \cdots \| \text{out}_u.$$  

Upper-bounding leakage from the output. Since $|K_\text{bad}|$ and $|K_{\theta,i}|$ are at most $k \in \text{poly}(\kappa)$, we can safely assume that the total number of bits in $\text{out}$ is

$$2 \lg |K_\text{bad}| + \sum_{i=1}^u \lg |K_{\theta,i}| \leq (2 + \lg \lg \kappa) \lg \text{poly}(\kappa) \leq \kappa.$$  

Distribution of $R$ and its min-entropy. We first consider the original distribution on $P_2$’s secret set $R$ to be the uniform distribution over all sets of size $n_R$ with each element is chosen from a universe $U$. The universe $U$ has size $\ell \cdot n_R$ with $\ell \geq 4(n_R)^3$.

Now choose, uniformly at random, a partition $\{U_1, \ldots, U_{n_R}\}$ of $U$ where each $|U_j| = \ell$ such that the element in the $j$th slot of $R$ belongs to $U_j$. These universes $\{U_1, \ldots, U_{n_R}\}$ are leaked in the analysis.

Let $D$ denote the original distribution over the set $R$, but conditioned on the leaked information $\{U_1, \ldots, U_{n_R}\}$. The distribution $D$ is equivalent to a distribution over streams of $n_R$ elements, where the element in the $i$-th slot is chosen uniformly at random from $U_i$. Therefore, $D$ has min-entropy $n_R \lg \ell$.

We additionally consider arbitrary leakage $f(R) = \gamma$ of length $L$. $L$ is set such that:

$$n_R \lg \ell - L \geq \frac{8n_R}{9} \lg \ell + 2n_R$$

Available iterations in a bundle. For a fixed set $Z \subseteq R$, we say that a set of iterations in the $i$th bundle $K_{\theta,i}$ is available with respect to $Z$ if there are no edges from $Z$ to that set. More formally, consider a graph $G \leftarrow \text{MinhashG}_{H_1}(A, I, x^*, H_2)$ and letting $G = (X, Y, E)$, we define

$$\text{Avail}_G(K_{\theta,i}, Z) := \{ j \in K_{\theta,i} : \forall z \in Z : (z, j) \notin E \}.$$  

Intuitively, no elements in the fixed set $Z$ can be minimum hash value in the $j$th iteration. In this sense, the iteration $j$ is still available for other elements than those in $Z$ to become the winner of delivering the minimum hash value.

Existence of a good bundle. We now describe a process for identifying a “good” bundle of iterations in the sense that given the fixed hash, the distribution $D$ (after the leakage) satisfies the DP-like property. We show that with all but negligible probability, process $\text{IsAGoodBundle}$ succeeds on at least one bundle $K_{\theta,i}$ where $i \in [u]$. 

37
Recall that \( \Pr[\text{SUCCESS}] \) with high probability.

Proof. Let \( D_1 := D \mid \text{out}_x \). In other words, \( D_1 \) is the distribution \( D \) conditioned on the output vector \( \text{out}_x \). If \( D_1 \) has min-entropy less than \( n_R \log(\ell) - L - \kappa \), then output \( \text{FAIL}_1 \) and terminate.

We note that by applying \[14\] Lemma 2.2, the average min-entropy of \( D \mid \text{out}_x \) is at least \( n_R \log(\ell - L - \kappa) \), which implies that the min-entropy of \( D \mid \text{out}_x \) is at least \( \frac{n_R \log(\ell - L - 2\kappa)}{2} \geq \frac{R_n}{2\kappa} \log(\ell) + n_R \) with probability \( 1 - 2^{-\kappa} \), assuming that \( n_R \geq 2\kappa \). Therefore, the probability that the process outputs \( \text{FAIL}_1 \) is at most \( 2^{2\kappa} \leq \text{negl}(\kappa) \).

Due to Lemmas \[8\] and \[9\] there is a leakage function \( f_G \) which leaks \( V = \{j_1, \ldots, j_{n_R}\} \) such that with all but negligible probability there exists a distribution with the Geometric Collision Property over sets \( R' = \{x_j \in R : j \in V\} \). If there is no such distribution, output \( \text{FAIL}_2 \) and terminate. The leakage function also leaks \( T = \text{Avail}_G(K_{\theta,i}, R \setminus R') \). Let \( D_{2,i} := D_1 \mid f_G \).

If it holds \( |T| \leq \frac{k_m}{10} |K_{\theta,i}| \), output \( \text{FAIL}_3 \) and terminate. Let \( k_v = |T| \).

Using \( G \) constructed above, compute

\[
D_{T,r}(D_{2,i}) = \Pr_{R' \sim D_{2,i}}[I_{R', r}].
\]

Check if there are \( a \) and \( b \) satisfying the following conditions:
- For \( r \notin [a + \log \log \kappa, b] \), \( D_{T,r}(D_{2,i}) \) is negligible in \( \kappa \).
- For \( r \in [a, b] \), it holds that \( e^{-r/3} E^{a_{n_R}} \leq D_{T,r}(\tilde{D}) \leq e^{r/3} E^{a_{n_R}} \).

Output \( \text{FAIL}_4 \) and terminate, if the above check fails.

6. Output SUCCESS.

SUCCESS with high probability. We already argued that \( \text{FAIL}_1 \) and \( \text{FAIL}_2 \) occur with negligible probability. We now argue that \( \text{FAIL}_3 \) occurs with negligible probability. Recall \( k_v = |K_{\theta,i}| \).

Lemma 14. Fix \( H_1, A, I, x^* \), and consider choosing \( H_2 \) and constructing \( G \leftarrow \text{Minhash}_{G_{H_1}}(A, I, x^*, H_2) \). Then, it holds that

\[
\Pr_{H_2}[|T| \leq k_v / 10] \leq \text{negl}(\kappa).
\]

Proof. Let \( n = n_R \) and \( n' = n_R \) for brevity of notation. Recall that \( n' = n/3 \). Let \( X_j \) be an indicator variable that represents whether there is an edge from \( (n - n') \) nodes to iteration \( j \). Therefore, we have

\[
\Pr_{H_2}[|T| = r] = \Pr_{H_2}\left[\sum_{j=1}^{k_v} X_j = k_v - r\right].
\]

Recall that \( p_j \leq 1 - (\eta - \theta)^{1/n'} \) and \( \Pr[X_j = 1] = 1 - (1 - p_j)^{n - n'} \leq 1 - (\eta - \theta)^{n - n'} = 1 - (\eta - \theta)^2 \leq 1 - (2/5)^2 \). Therefore, we have

\[
m := \mathbb{E}\left[\sum_{j=1}^{k_v} X_j\right] \leq k_v \cdot (1 - (\eta - \theta)^2) \leq 0.84 k_v
\]
Using the Chernoff bound and due to $k_b \in \Omega(\kappa)$, we have

$$
\Pr_{H_2} \left[ |T| \leq \frac{k_b}{10} \right] = \Pr_{H_2} \left[ \sum_{j=1}^{k_b} X_j \geq \frac{9}{10} k_b \right] \leq \exp \left( -\frac{(0.9k_b - m_0)^2}{2m_0} \right) = \exp(-\Omega(\kappa)). \square
$$

We now analyze $\text{FAIL}_{4,i}$. In particular, by Lemma 7 for all $i \in [u]$, conditioned on $\text{FAIL}_{1,i}$, $\text{FAIL}_{2,i}$, $\text{FAIL}_{3,i}$ not occurring, let

$$
p_4 := \Pr_{H_1, H_2} [\text{IsAGoodBundle}(i, \text{out}, A, I, x^*, H_1, H_2) = \text{FAIL}_{4,i}].
$$

Then, we have $p_4 \in O(k_v \log^3(\kappa)/(n_R)^{0.5})$.

Further, observe that conditioned on $\text{FAIL}_{1,i}$, $\text{FAIL}_{2,i}$, $\text{FAIL}_{3,i}$ not occurring, the process outputs $\text{FAIL}_{4,i}$ independently of $\text{out}_{-i}$, since the hash values in $H_2$ for any iteration are chosen independently of those for the other iterations. Using the above, since $k_v/\sqrt{n_R} = O(1/\sqrt{\kappa})$, we have the following:

The process $\text{IsAGoodBundle}$ outputs SUCCESS for at least one bundle with probability $1 - p_4^u = 1 - \negl(\kappa)$.

**Noise distribution.** We define a noise distribution $\Phi$ and give an analysis of the hockey stick divergence of $\Phi(r)$ and $\Phi(r - \log \log(\kappa))$.

**Definition 9 (Noise distribution $\Phi$).** We define $\Phi(r)$ as follows:

- Choose $H_1$ and $H_2$ randomly.
- Let $i^* \in [u]$ be the index to the bundle that $\text{IsAGoodBundle}$ outputs SUCCESS.
- For $r \in [0, k_v]$, output $D_{T,r}(D_{2,i^*})$, where $T = \text{Avail}_G(K_{\emptyset,i^*}, R \setminus R')$.
- For $r \notin [0, k_v]$, $\Phi(r) := 0$.

**Lemma 15.** The hockey stick divergences $D_{r,v}^{hs}(\Phi(r), \Phi(r - \log \log(\kappa)))$ and $D_{r,v}^{hs}(\Phi(r - \log \log(\kappa)), \Phi(r))$ are both negligible in $\kappa$.

**Proof.** For brevity, for any $r$, denote $D_r := D_{T,r}(D_{2,i^*})$. Conditioned on $\text{IsAGoodBundle}$ outputting SUCCESS with input $\text{out}_{-i}$, we have $a$ and $b$ such that for $r \in [a + \log \log \kappa, b]$,\n
$$
e^{-\epsilon} \leq \frac{e^{-\epsilon/3} E_{k_v,r}^{\text{hs}}}{e^{\epsilon/3} E_{k_v,r - \log \log(\kappa)}^{\text{hs}}} \leq \frac{D_r}{D_r - \log \log(\kappa)} \leq \frac{e^{\epsilon/3} E_{k_v,r}^{\text{hs}}}{e^{-\epsilon/3} E_{k_v,r - \log \log(\kappa)}^{\text{hs}}} \leq e^{\epsilon}.
$$

The first and last inequalities are from Lemma 4. The second and third inequalities are from the condition that the process outputs SUCCESS. The hockey stick divergence $D_{r,v}^{hs}(\Phi(r), \Phi(r - \log \log(\kappa)))$ is therefore at most

$$
\sum_{r \not\in [a + \log \log \kappa, b]} D_r \leq k_v \cdot \negl(\kappa) = \negl(\kappa). \square
$$

Similarly, $D_{r,v}^{hs}(\Phi(r - \log \log(\kappa)), \Phi(r))$ is also $\negl(\kappa)$. 39
Putting it all together. Let \( c \) be the final count produced by running protocol \( \pi_{PH} \). We consider the probabilities

\[
Pr_{H_1, H_2, D}[c \mid B_{+x^*}^\star] \quad \text{and} \quad Pr_{H_1, H_2, D}[c \mid B].
\]

We consider only runs of the protocol that yield \( c \) and for which there exists some \( i^* \in [u] \) such that the process \( \text{IsAGoodBundle} \) returns \text{SUCCESS} given \( out_{-i^*} \) as input. We just have argued that such an \( i^* \) exists with all but negligible probability.

Further, we consider only runs of the protocol for which \( out_{+}^\star - out_{bad} \leq s = \lg \lg(\kappa) \). By Lemma 2, this also occurs with all but negligible probability, We will also leak \( k = \lvert \text{Avail}(K_{\theta,i^*}, R \setminus R') \rvert \).

Conditioned on the above events, by the definition of the distribution \( \Phi \), the value \( out_i \) contributes \( (k - r) \) to the final count \( c \) with probability \( p = \Phi(r) \).

Recall that every iteration \( j \) in \( K_{\theta,i^*} \) is good, which means \( \min h_j(A) = \min h_j(I) \), potentially contributing to the output.

Therefore, assuming none of bad events occur (which happens with overwhelming probability), by applying Lemma 15, the probability that the ratio of probabilities of a certain output \( out \) for \( B_{+x^*}^\star \) and \( B \) is not contained in \( [e^{-\epsilon}, e^{\epsilon}] \) is \( \text{negl}(\kappa) \), and therefore we conclude that the protocol satisfies the DDP security.

E Proof of Lemma 7

Notations. When considering the probability of \( D_{T,r}(\tilde{D}) \) and \( I_{R,T,r} \) over the choice of \( H_2 \), then the identity of \( T \) doesn’t matter except for its size \( \hat{k} = |T| \). Therefore, in this case, we will simply use \( D_{\hat{k},r}(\tilde{D}) \) and \( I_{R,\hat{k},r} \). Moreover, when it is clear from the context, we will sometimes omit \( \hat{k} \) and \( \tilde{D} \) and say \( E'_r = E'_r^n \) and \( I'_{r,r} = I'_{\hat{k},r,r} \), and \( D_r = D_{\hat{k},r}(\hat{D}) \).

We first show the following lemma holds.

**Lemma 16.** Let \( D \) be a distribution over sets of size \( n'_R \) with geometric collision property. Fix \( H_1 \) and consider \( \hat{k}, \theta, a, b \) specified in Lemma 7 with the same requirements. Then, we have the following:

Case 1: If \( r \not\in [a+s, b] \), then we have

\[
Pr_{H_2}[D_{\hat{k},r}(\tilde{D}) \leq \text{negl}(\kappa)] \geq 1 - \text{negl}(\kappa).
\]

Case 2: If \( r \in [a, b] \), then we have

\[
Pr_{H_2}[e^{-\epsilon/3}E_{\hat{k},r}^{n'_R} \leq D_{\hat{k},r}(\tilde{D}) \leq e^{\epsilon/3}E_{\hat{k},r}^{n'_R}] \geq 1 - (e^{\epsilon/3} - 1)^{-2} \cdot \frac{16 \log^3(\kappa)}{\sqrt{n'_R}}.
\]

Then, Lemma 7 follows by taking a union bound over different cases of \( r \). \( \square \)
E.1 Proof of Lemma 16

We also define \( \rho(R') := \Pr_{R' \sim \tilde{D}} [R'] \).

**Proof for Case 1.** We first consider Case (1). By applying the Case (1) of Corollary 1, we have \( E_r^{n_R} \in \negl(\kappa) \). Given \( E_r^{n_R} \in \negl(\kappa) \), we show

\[
\Pr_{H_2} [D_r(\tilde{D}) \leq \negl(\kappa)] \geq 1 - \negl(\kappa).
\]

Recall that \( D_r(\tilde{D}) = \sum_{R'} \rho(R') \cdot I_{R', r} \). Assume toward the contradiction that the negation of the statement holds. This means there are polynomials \( p \) and \( q \), and a collection \( \text{Heavy} \) of \( R' \)’s such that

\[
\Pr_{H_2} \left[ \sum_{R' \in \text{Heavy}} \rho(R') \cdot I_{R', r} \geq 1/p(\kappa) \right] \geq 1/q(\kappa).
\]

Note that \( \sum_{R' \in \text{Heavy}} \rho(R') \geq 1/p(\kappa) \). Now, since \( \tilde{D} \) and \( H_2 \) are independent, the above implies that

\[
\sum_{R' \in \text{Heavy}} \rho(R') \Pr_{H_2} [I_{R', r}] \geq \frac{1}{p(\kappa)q(\kappa)}.
\]

However, considering that \( \Pr_{H_2} [I_{R', r}] = E_r^{n_R} \), which is negligible, the above is a contradiction.

**Proof for Case 2.** We will bound \( D_r = \sum_{R'} \rho(R') \cdot I_{R', r} \) using Chebyshev inequality. For this, we would like to bound the variance of \( D_r \).

We start with showing the following lemma, which will allow us to ignore the tail when we bound the variance. Below, the value \( z \) will correspond to the size of the intersection of the two sets \( R'_i \) and \( R'_j \).

**Lemma 17.** Fix \( H_1 \). Consider a graph \( G \leftarrow \text{Minhash}_{H_1}(A, I, x^*, H_2) \). Consider any set \( T \) iterations in \( G \) such that \( |T| = \hat{k} \). Let \( Z \) be a set of left nodes in \( G \) such that \( |Z| \leq n'_R \). Let \( z = |Z| \). Consider the probability (over the choice of \( H_2 \)) that \( Z \) has more than \( z \lg \lg \kappa \) outgoing edges in \( G \). This probability is negligible in \( \kappa \).

**Proof (Lemma 17).** Let \( p = 1 - (\eta - \theta)^{1/n'_R} \). We first show that \( p \leq 1/n'_R \). Recall \( \theta \leq 1/10 \), which implies \( e^{-\theta} \leq 1/2 - \theta = \eta - \theta \). Therefore, we have \( (1 - 1/n'_R)^{n'_R} \leq e^{-\theta} \leq \eta - \theta \), so \( 1 - 1/n'_R \leq (\eta - \theta)^{1/n'_R} \). Therefore, we have \( 1 - (\eta - \theta)^{1/n'_R} \leq 1/n'_R \).

Let \( \text{Edges}(Z, T) \) be the set of edges from \( Z \) to \( T \). Over the choice of \( H_2 \), the probability that each pair in \( Z \times T \) forms an edge is at most \( p \). Therefore, we can simply use a Binomial distribution to bound the probability. In particular,
with \( t = \lg \lg \kappa \) we have

\[
\Pr_{H_2}[|\text{Edges}(Z,T)| \geq zt] \leq \Pr[B(z\hat{k}, p) \geq zt]
\]

\[
\leq \frac{z\hat{k}}{zt} \cdot p^{zt}
\]

\[
\leq \frac{z\hat{k}}{zt} \cdot (1/n'_R)^{zt}
\]

\[
\leq \left( \frac{e \cdot \hat{k}}{n'_R \cdot t} \right)^{zt}.
\]

Since \( n'_R \) is much larger than \( \hat{k} \), the above probability becomes negligible in \( \kappa \).

Now we prove the following lemma towards bounding the variance of \( D_r \).

**Lemma 18.** Fix \( H_1 \). We set the parameters for \( \hat{k}, a \) and \( b \) as stated in Lemma 7. Let \( R'_i, R'_j \) be sets of nodes on the left of size \( n'_R \) such that with \( |R'_i \cap R'_j| = z \).

Let \( \zeta = z \lg \lg \kappa \). Then for all \( a \leq r \leq b \), we have

\[
\Pr_{H_2}[I_{R'_i,r} \land I_{R'_j,r}] = \mathbb{E}_{H_2}[I_{R'_i,r} \cdot I_{R'_j,r}]
\]

\[
\leq \left( 1 + \frac{\zeta \cdot (e^{\zeta/3} + 1)}{\eta - \tilde{d}^{\zeta/k/n'_R}} \right) \left( E^{n'_R} \right)^2.
\]

**Proof.** Fix \( R'_i, R'_j \) with \( |R'_i \cap R'_j| = z \). Let \( Z = R'_i \cap R'_j \) and \( X = R'_i - Z \). Then, we have

\[
\Pr_{H_2}[I_{R'_i,r} \land I_{R'_j,r}] = \sum_{m=0}^{r} \Pr[I_{X,m} \land I_{Z,0} \land I_{R'_j,r}]
\]

\[
\leq \sum_{m=0}^{r-\zeta} \Pr[I_{Z,0} \land I_{Z,0} \land I_{R'_j,r}]
\]

\[
= \sum_{m=\zeta}^{r} \Pr[I_{Z,m} \land I_{R'_j,r}]
\]

\[
\leq \negl(\kappa) + \sum_{m=r-\zeta+1}^{r} \Pr[I_{X,m} \land I_{R'_j,r}]
\]

\[
= \negl(\kappa) + E^{n'_R} \cdot \sum_{m=r-\zeta+1}^{r} \Pr[I_{X,m}].
\]

The second inequality holds due to Lemma 8. It is left to bound \( \Pr[I_{X,m}] \) for \( m \in (r - \zeta, r] \). We observe that the following holds:

\[
\Pr_{H_2}[I_{X,m}] = \Pr[I_{R'_i,m} \mid I_{Z,0}].
\]
In other words, the event that \( X \) contributes to noise pattern \( m \) is equivalent to the event that \( R'_i \) contributes to \( m \) conditioned on the intersection having no contribution.

Therefore, we have

\[
\Pr_{H_2}[I_{X,m}] = \frac{\Pr[I_{R'_i,m} \land I_{Z,0}]}{\Pr[I_{Z,0}]} \leq \frac{\Pr[I_{R'_i,m} \land I_{Z,0}]}{\eta^{-z_k/n'_{R}}} = \frac{\Pr[I_{R'_i,m}]}{\eta^{-z_k/n'_{R}}} = \frac{E_{m}^{n_{R'}}}{\eta^{-z_k/n'_{R}}}.
\]

We now bound \( E_{m}^{n_{R'}} \) for \( m \in (r - \zeta, r] \). Let \( m^* := \arg\max_m \{ E_{m}^{n_{R'}} : m \in (r - \zeta, r] \} \). Using Corollary 1 we have

\[
E_{m^*}^{n_{R'}} \leq (e^{\epsilon/3} \cdot E_{r}^{n_{R'}} + \text{negl}(\kappa)).
\]

Therefore, we have

\[
\Pr_{H_2}[I_{R'_i,r} \land I_{R'_j,r}] \leq \text{negl}(\kappa) + E_{r}^{n_{R'}} \cdot \sum_{m=r-\zeta+1}^{r} \Pr[I_{X,m}] \\
\leq \text{negl}(\kappa) + \zeta \cdot E_{r}^{n_{R'}} \cdot \Pr[I_{X,m^*}] \\
= \text{negl}(\kappa) + \zeta \cdot E_{r}^{n_{R'}} \cdot \frac{E_{m^*}^{n_{R'}}}{\eta^{-z_k/n'_{R}}} \\
= \text{negl}(\kappa) + \zeta \cdot E_{r}^{n_{R'}} \cdot \frac{e^{\epsilon/3} E_{r}^{n_{R'}} + \text{negl}(\kappa)}{\eta^{-z_k/n'_{R}}} \\
\leq \left(1 + \frac{\zeta \cdot (e^{\epsilon/3} + 1)}{\eta^{-z_k/n'_{R}}} \right) \left( E_{r}^{n_{R'}} \right)^2 \square
\]

**Lemma 19.** We set the parameters for \( H_1, \hat{k}, \alpha \) and \( b \) as stated in Lemma 7. Let \( D \) be a distribution with the geometric collision property. Then, for every \( a \leq r \leq b \), we have

\[
\text{Var}_{H_2}[D_r] \leq \frac{16 \log^3(\kappa)}{\sqrt{n_{R}}} \left( E_{k,r}^{n_{R'}} \right)^2.
\]

**Proof.** Consider any \( r \in [a, b] \). Recall that \( D_r := \sum_{R' \in \text{Supp}(\tilde{D})} \rho(R') \cdot I_{R',r} \).
\[ \text{Var}_{H_2}[D_r] = \sum_{R_i', R_j'} \rho(R_i') \cdot \rho(R_j') \cdot \left( \mathbb{E}[I_{R_i', r} \cdot I_{R_j', r}] - \mathbb{E}[I_{R_i', r}] \cdot \mathbb{E}[I_{R_j', r}] \right) \]
\[ \leq \sum_{R_i', R_j' : |R_i' \cap R_j'| \geq 1} \rho(R_i') \cdot \rho(R_j') \cdot \mathbb{E}[I_{R_i', r} \cdot I_{R_j', r}] \]
\[ = \sum_{z=1}^{n_R} \Pr_{R_i', R_j' \sim \mathcal{D}} \left[ |R_i' \cap R_j'| = z \right] \cdot \mathbb{E}[I_{R_i', r} \cdot I_{R_j', r}] \]
\[ \leq \sum_{z=1}^{n_R} \left( \frac{1}{\sqrt{n_R}} \right)^z \cdot \left( 1 + \frac{\zeta \cdot (\epsilon^{\zeta/3} + 1)}{\eta \theta^{3k/n_R}} \right) \cdot \left( E_{r}^{n_R} \right)^2 \]
\[ \leq \sum_{z=1}^{n_R} \left( \frac{1}{\sqrt{n_R}} \right)^z \cdot \left( \zeta \cdot \frac{\epsilon^\zeta + 2}{(2/5)^{\zeta/3}} \right) \cdot \left( E_{r}^{n_R} \right)^2 \]
\[ \leq \sum_{z=1}^{n_R} \left( \frac{1}{\sqrt{n_R}} \right)^z \cdot \left( 8^{\zeta+1} \right) \cdot \left( E_{r}^{n_R} \right)^2 \]
\[ = 8 \cdot \left( E_{r}^{n_R} \right)^2 \cdot \sum_{z=1}^{n_R} \left( \frac{10 \cdot 3 \cdot \zeta}{\theta^{3k/n_R}} \right)^z \]
\[ \leq 16 \log^3(\zeta) \left( E_{r}^{n_R} \right)^2. \]

The first inequality holds because if \( R_i' \) and \( R_j' \) are disjoint, then \( I_{R_i', r} \) and \( I_{R_j', r} \) are independent over the choice of \( H_2 \), and the relevant terms are canceled out. The second inequality is due to the geometric collision property of \( \mathcal{D} \) and Lemma 18. The third inequality holds with \( \epsilon \leq 3 \) since \( \theta < 1/10 \) and \( k \) is much smaller than \( n_R \). □

Finally, by Chebyshev, we have that for all \( a \leq r \leq b \),
\[ \Pr_{H_2} \left[ D_r \notin \left[ e^{-\epsilon/3}(E_{k,r}^{n_R}), e^{\epsilon/3}(E_{k,r}^{n_R}) \right] \right] \leq \Pr \left[ |D_r - E_{k,r}^{n_R}| \geq (1 - e^{-\epsilon/3}) \cdot E_{k,r}^{n_R} \right] \]
\[ \leq \frac{\text{Var}[D_r]}{(1 - e^{-\epsilon/3})^2 \cdot (E_{k,r}^{n_R})^2} \]
\[ \leq \frac{16 \log^3(\zeta)}{(1 - e^{-\epsilon/3})^2 \cdot n_R}. \]

**F Distribution with the Geometric Collision Property**

In this section, we show that the distribution \( \mathcal{D}_{2,i} \) described in process **IsAGoodBundle** possesses the geometric collision property. Note that the receiver’s input is chosen from the following distribution:
For each $i \in \{n_R, n_B\}$, choose $x_i^B$ uniformly at random from the universe $U_i$. We assume that for $i \neq j$, $U_i$ and $U_j$ are disjoint with the same cardinality $|U_i| = |U_j| := \ell$.

Indeed, the above distribution has the geometric collision property as long as $\ell \geq n_R \cdot \sqrt{n_R}$ (recall $n_R = n_B - n_f$). When considering two random sets $R_0'$ and $R_0'$ of size $n_R' = n_R/3$ where their elements are from the above distribution, each position $i$ will have high min-entropy, even conditioned on the previous elements.

The main issue is that we need to deal with the leakage stemming from the fact that the hash functions are public. Therefore, we need to consider a more general class of distributions that captures the leaked version of the above distribution and show that they still possess the geometric collision property.

**Strong chain rule for a special case: achieving flatness through clustering.** Fortunately, a stronger version of the chain rule is known to hold for a special leakage pattern, i.e., when elements are conditioned in order \([11]\): very roughly speaking, for every $i$, the min-entropy of $R_i | (R_1, \ldots, R_{i-1})$ is essentially the same as the min-entropy of $(R_1, \ldots, R_i)$ minus the min-entropy of $(R_1, \ldots, R_{i-1})$ at the sacrifice of an additional small leakage, which is called a *spoiling leakage.*

They achieve this by grouping possible sequences with a similar distributional characteristic into the same cluster. Then, in every cluster, the distribution of sequences condition on that cluster will be essentially flat. Now, the spoiling leakage corresponds to the cluster identifier. By making every cluster contain sufficiently many sequences (leading to sufficient min-entropy due to flatness), the total number of clusters can be small (leading to a short spoiling leakage).

**Notes on notations.** For brevity, in this section, we omit the subscript from $n_R$, i.e., we denote $n = n_R$. For any sequence of random variables $R = R_1, \ldots, R_n$ (for the secret input $R$), we denote $R_{<i} = R_1, \ldots, R_{i-1}$ and $R_{\geq i} = R_1, \ldots, R_i$. Likewise, we extend such subscript notations and use $R_{>i}$ and $R_{\leq i}$. We use lower case $r = r_1, \ldots, r_n$ to denote the actual set/sequence.

**Strong chain rule for our setting.** We first adapt the result in \([11]\) into our setting, in which itself needs a significant amount of modification. Then, we argue that a sufficient number of elements still have high min-entropy, even conditioned on the previous elements. Finally, we show that these high min-entropy (conditioned) elements provide the geometric collision property.

**Theorem 5 (Block structures with few bits spoiled in our setting).** We consider a min-hash graph $G = (X, Y, E)$ constructed from $\text{Minhash}_{G_{H_1}}(A, I, x^*, H_2)$, while focusing on a single bundle $K_{\theta, \epsilon}$ of iterations.

Let $U = U_1 \times \cdots \times U_n$ be a fixed universe and $R = (R_1, \ldots, R_n)$ be a sequence of (possibly correlated) random variables where each $R_i$ is over $U_i$ (and all are disjoint) and $|U_i| = \ell$ for all $i$. Then, for any $\epsilon \in (0, 1)$ and any $\delta > 0$, there exists a spoiling leakage function $f_G(R)$ that satisfies the following properties.
1. It holds that $\Pr[R_i[f(R) = \perp]] \leq cn$.
2. Let $\text{Im}(f)$ be the set of images of $f$. Every $y \in \text{Im}(f) \setminus \{\perp\}$ specifies two disjoint sets $V$ and $W$ such that $V \cup W = [n]$.
3. Conditioned on any $y \in \text{Im}(f) \setminus \{\perp\}$, for every $i \in V$, every element in distribution $R_{i \mid R_{<i}}$ has low probability weight, i.e.,
\[
\forall y \in \text{Im}(f) \setminus \{\perp\}, \forall r \text{ s.t. } f(r) = y, \forall i \in V: \Pr\left[R_i = r_i \mid R_{<i} = r_{<i}, y\right] \leq \frac{2^5}{n^{1.5}}.
\]
4. Conditioned on any $y \in \text{Im}(f) \setminus \{\perp\}$, for every $i \in W$, it holds that $R_{i \mid R_{<i}}$ has small support size, i.e.,
\[
\forall y \in \text{Im}(f) \setminus \{\perp\}, \forall r \text{ s.t. } f(r) = y, \forall i \in W: |\{r_i : \Pr[R_i = r_i|R_{<i} = r_{<i}, y]\}| \leq 2^{6} \cdot n^{1.5}.
\]
5. $|\text{Im}(f)| \leq n \cdot (2e)^{n/2} \cdot \frac{(n + \log n)}{n} \cdot (2(\log(\ell) + \log(1/\epsilon))/\delta)^{\delta}$.
6. $\text{Avail}(K_{\theta_s}, R_W)$ can be computed from $f(R)$, where $R_W := \{R_i : i \in W\}$.

F.1 Proof of Theorem 5

By following the general idea of [41], we will build clusters, and the spoiling leakage will be the cluster identifier. However, we will slightly change the way we build clusters.

**Condition 1.** Throughout our proof, we let $\Pr[r_i]$ denote $\Pr[R_i = r_i]$ for brevity, whenever the referred random variable is clear. Before forming the clusters, we will first like to exclude all sequences $r \in \mathcal{U} = U_1 \times \cdots \times U_n$ having a very small probability $\Pr[R_i = r_i \mid R_{<i} = r_{<i}] < \epsilon/\ell$ for any $i \in [n]$ and only consider the remaining $\mathcal{U}' \subset \mathcal{U}$. Specifically, we let $f(r) = \perp$ for all $r \notin \mathcal{U}'$. As we will see later, this probability lower bound is vital to upper bound $|\text{Im}(f)|$.

**Claim.** Let $\mathcal{U}'$ be the set containing all the sequences $r$ such that $\Pr[R_i = r_i \mid R_{<i} = r_{<i}] \geq \epsilon/\ell$ for all $i \in [n]$. Then, we have $\Pr[r \in \mathcal{U}'] \geq 1 - cn$.

**Proof.** For each $i \in [n]$, and any $r_{<i} \in U_1 \times \cdots \times U_{i-1}$, we have
\[
\sum_{u \in U_i : \Pr\left[R_i = u \mid R_{<i} = r_{<i}\right] < \epsilon/\ell} \Pr[R_i = u \mid R_{<i} = r_{<i}] < \sum_{u \in U_i} \frac{\epsilon/\ell}{n} = \epsilon.
\]
Therefore, using a union bound across all $i \in [n]$, we have
\[
\Pr[r \notin \mathcal{U}'] \leq \epsilon \cdot n.
\]

**Building clusters.** For each $r \in \mathcal{U}'$, we describe how to compute $f(r) = (f_1(r), f_2(r), \ldots, f_n(r))$, which will serve as the cluster identifier. Let $r(a)$ denote a rounding function that rounds $a$ to the closest multiple of $\delta/2$. We say $a \approx_r a'$ if $r(a) = r(a')$. 

46
For each $r$, do the following:
1. Let $f_{>n}(r) = \perp$ for any $r$, and initialize $W = \emptyset$.
2. For $i = n, \ldots, 1$, do the following:
   (a) Let $sp_1^i(r)$ denote the surprise of the $i$th element of $r$. More formally,
   \[
   sp_1^i(r) = -\log \Pr_{R} [R_i = r_i \mid R_{<i} = r_{<i}, f_{>i}(R) = f_{>i}(r)].
   \]
   This surprise measure represents how rare and surprising the event $r_i$ is, conditioned on $r_{<i}, f_{>i}(r)$. In a sense, we will group sequences with similar surprises into a cluster.
   (b) Let $sp_2^i(r)$ denote the surprise of the sequences with a similar surprise level in aggregate.
   \[
   sp_2^i(r) = -\log \Pr_{R} [sp_1^i(R) \approx sp_1^i(r) \mid R_{<i} = r_{<i}, f_{>i}(R) = f_{>i}(r)].
   \]
   Note $sp_1^i(r) \geq sp_2^i(r)$, since at least sequence $r$ has $sp_1^i(r)$ and possibly more points may approximately share the surprise. Note also that $sp_2^i(r)$ is a deterministic function of $sp_1^i(r), r_{<i}, f_{>i}(r)$.
   (c) If $r(sp_1^i(r)) - r(sp_2^i(r)) \geq 1.5 \log(n)$ then let $f_i(r) = (r(sp_1^i(r)), \text{true})$.
   (d) Otherwise, let $f_i(r) = (r(sp_1^i(r)), \text{false}, H_i)$ and add $i$ to $W$.
   Here, $H_i$ is defined as $N(\{r_i\}) \setminus N(r_W)$, where $N$ refers to the neighbors (restricted to $K_{\theta_*}$) of the input set of nodes in $G$. In other words, $H_i$ contains the iterations newly covered by element $r_i$; any iterations previously covered by $r_W$ are ruled out in $H_i$.
   In this way, we can reduce the length of the cluster identifier.
3. Set $f(r) = f_1(r), \ldots, f_n(r)$. Set $V = [n] \setminus W$.

**Conditions 2 and 3.** Condition 2 follows from how $V$ is computed in step 3. We now show that condition 3 holds. In particular, $\forall y \in \text{Im}(f(\cdot)) \setminus \{\perp\}, \forall r$ s.t. $f(r) = y, \forall i \in V$ we have
\[
\Pr[r_i \mid r_{<i}, y] = \Pr[r_i \mid r_{<i}, y_{\geq i}] = \frac{\Pr[r_i \wedge r_{<i} \wedge y_{\geq i}]}{\Pr[r_{<i} \wedge y_{\geq i}]} = \frac{\Pr[r_i \wedge r_{<i} \wedge y_{\geq i}]}{\Pr[r_{<i} \wedge y_{\geq i}] \Pr[y_i \mid r_{<i} \wedge y_{\geq i}]}
\]

The first equality is due to $y_{<i}$ being a deterministic function of $r_{<i}, y_{\geq i}$. Similarly, the nominator of the final fraction is due to $y_i$ being a deterministic function of $r_{<i}, y_{\geq i}$. Moreover, $y_{i,2}$ (i.e., $\text{true}$) can be deterministically computed from $y_{i,1}$ (i.e., $r(sp_1^i(r)))$, $r_{<i}, y_{>i}$. Therefore, the above is equal to
\[
\frac{\Pr[r_i \wedge r_{<i} \wedge y_{>i}]}{\Pr[y_{i,1} \mid r_{<i} \wedge y_{>i}] \Pr[r_{<i} \wedge y_{>i}]} = \frac{\Pr[r_i \mid r_{<i} \wedge y_{>i}]}{\Pr[y_{i,1} \mid r_{<i} \wedge y_{>i}]} = \frac{2^{-sp_1^i(r)}}{2^{-sp_1^r(r)}} \leq \frac{2^4}{n^{1/5}}. \tag{5}
\]

The last inequality holds since $i \in V$, $r(sp_1^i(r)) - r(sp_1^i(r)) \geq 1.5 \log(n)$.
**Condition 4.** For \( r, y, i \) as quantified in the theorem statement, we have
\[
|\{ r_i : \Pr[R_i = r_i | R_{<i} = r_{<i}, y] \geq 0 \}| = |\{ r_i : \Pr[R_i = r_i \land R_{<i} = r_{<i} \land y] \geq 0 \}| = |\{ r_i : \Pr[R_i = r_i \land R_{<i} = r_{<i} \land y_{i,1}, y_{i,2}, y_{>i}] \geq 0 \}| \leq |\{ r_i : \Pr[R_i = r_i | R_{<i} = r_{<i}, y_{i,1}, y_{i,2}, y_{>i}] \geq 0 \}|
\]
By a similar argument as above, for all \( r_i \) such that
\[
\Pr[R_i = r_i | R_{<i} = r_{<i}, y_{i,1}, y_{i,2}, y_{>i}] \geq 0,
\]
it holds
\[
\Pr[R_i = r_i | R_{<i} = r_{<i}, y_{i,1}, y_{i,2}, y_{>i}] = \frac{\Pr[r_i | R_{<i} = r_{<i} \land y_{i,1}] \Pr[y_{i,1} | r_{<i} \land y_{i,1} \land y_{>i}]}{= \frac{2^{-sp^1_i(r)}}{2^{-sp^2_i(r)}} \geq \frac{2-\delta}{n^{1.5}}
\]
where the inequality holds since \( i \in W \), we know that \( r(sp^1_i(r)) - r(sp^2_i(r)) \leq 1.5 \lg(n) \). This means that
\[
|\{ r_i : \Pr[R_i = r_i | R_{<i} = r_{<i}, y_{>i}] \geq 0 \}| \leq 2^\delta \cdot n^{1.5}.
\]

**Condition 5.** To bound \( |\text{Im}(f)| \), we first upper bound \( y_{i,1} \). Recall that \( \Pr[R_i = r_i | R_{<i} = r_{<i}] \geq \epsilon/\ell \) for all \( i \in [n] \) and \( r \in U' \). Therefore, \( \Pr[R_i = r_i | R_{<i} = r_{<i}, y] \geq \epsilon/\ell \) for all \( i \in [n] \), and \( \forall r \) such that \( f(r) = y \).

Therefore, for all \( r \in U', i \in [n] \), we have
\[
sp^1_i(r) \leq \lg(\ell) + \lg(1/\epsilon),
\]
which implies that \( y_{i,1} \) has at most \( 2(\lg(\ell)) + \lg(1/\epsilon)/\delta \) different possibilities.

To upper bound the number of possibilities of the remaining parts, it suffices to upper bound the number of choices for set \( W \) of size \( m \), as well as the number of possibilities for \( H_i \)'s in each slot \( i \in W \). Clearly, the former is \( \binom{n}{m} \). For the latter part, note that each iteration appears at most once over all \( m \) slots. Therefore, the problem becomes how we can assign \( k_b \) different iterations into \( m + 1 \) positions (with some positions possibly containing none) while assigning them to the \( m + 1 \)th position when they never appear in any slot of \( W \). This is a well-known problem of stars and bars with \( m + 1 \) variables and sum \( k_b \), which has \( \binom{m+k_b}{m} \) possibilities. Since we have \( k_b! \) different orderings for \( k_b \) iterations, the upper bound is \( \binom{m+k_b}{m} \cdot (k_b!) \). We have:
\[
|\text{Im}(f)| \leq \frac{(2(\lg(\ell) + \lg(1/\epsilon))/\delta)^n}{\left( \sum_{m=0}^{n} \binom{n}{m} \binom{m+k_b}{m} \cdot k_b! \right)} = \frac{(2(\lg(\ell) + \lg(1/\epsilon))/\delta)^n}{\left( \sum_{m=0}^{n} \binom{n}{m} \binom{m+k_b}{m} \cdot k_b! \right)} \leq n \cdot \frac{n}{n/2} \cdot \frac{(n+k_b)!}{n!} \cdot (2(\lg(\ell) + \lg(1/\epsilon))/\delta)^n \leq n \cdot (2\epsilon)^{n/2} \cdot \frac{(n+k_b)!}{n!} \cdot (2(\lg(\ell) + \lg(1/\epsilon))/\delta)^n.
\]
Condition 6. Finally, condition 6 follows from the definition of the clustering procedure. In particular, \( H_W = \bigcup_{i \in W} H_i \) contains all the iterations that \( r_W \) covers. The available set can be computed by \( K_{\theta,s} \setminus H_W \). This concludes our proof.

F.2 Generalization

It can be seen that in the above proof, the only properties that we used of the additional leakage \( H_i \) is that for \( i \in W \), \( H_i \) depends only on \( R_i, y_i \) and that the number of choices for the output of the sequence of leakages \( [H_i]_{i \in W} \) is bounded by some \( B \). Theorem 4 stated in Section 7 is restatement of Theorem 5 with respect to any such leakage function.

Note that the leakage functions \( \ell_i \) specified above can model leakage with respect to a random oracle \( h \), by letting \( \rho_i = h(R_i) \).

F.3 Towards Achieving Geometric Collision Property

Remember that we would like to show that when \( R \) is chosen uniformly at random from universe \( \mathcal{U} \) then the distribution of these \( n_R = n_B - n_I \) elements has the geometric collision property even with the leakage.

Towards this goal, in this section, by applying Theorem 5 to this distribution, we show that even with the leakage, there are at least \( n_R/3 \) elements that preserves enough min-entropy. In the next section, we show how these elements with sufficient min-entropy give the geometric collision property. For brevity, we let \( n := n_R \) in this subsection and \( n' := n_R = n_R/3 \).

Remark 1 (Getting rid of tiny parts). Similar to [41, Remark 2], we can further require that each cluster should have a probability that is “not too small”. Therefore, we define a new leakage function \( f' \) by substituting the \( \epsilon \) in the above theorem with \( \epsilon/2 \), and additionally letting \( f'(r) = \perp \) for all \( r \) such that \( y \in f(r) \) and \( \Pr_{R}[f(R) = y] < \epsilon n/(2|\text{Im}(f)|) \) (their total probability is at most \( \epsilon n/2 \)), we obtain the following:

\[ f' \] satisfies all conditions in Theorem 5. Additionally, \( \forall y \in \text{Im}(f') \), we have \( \Pr_{R}[f'(R) = y] \geq \epsilon n/(2|\text{Im}(f)|) \).

F.4 Proof of Lemma 8

In Lemma 8, by setting \( \ell \geq 4n^3 \) and assuming sufficient min-entropy of \( R \), we show that one can ensure more than \( 1/3 \) fraction of the blocks having min-entropy at least \( 1.5 \lg(n) \), upon leaking the outcome of \( f' \) and all previous blocks.

49
First notice that with all but $\epsilon n$ probability, $f'(R) \neq \bot$. Then, by setting $\delta = 1$, we have
\[
\log(|\text{Im}(f)|) \leq n \cdot (2\epsilon)^{n/2} \cdot (n + k_b) k_b \cdot (2(\log(\ell) + \log(1/\epsilon))/\delta)^n
\]
\[
= \left( \log(n) + n/2 \cdot \log(2e) \right) + k_b \cdot \log(n + k_b) + n \cdot (1 + \log(\log(\ell) + \kappa))
\]
\[
< 3n/2 + 2k_b \log n + n(2 + \log \kappa)
\]
\[
< 0.5n \log n
\]
for sufficiently large $n$ with $k_b = \Omega(\kappa)$ and $n/k_b^2 = \Omega(\kappa)$.

Combining the above with Remark 1, we have $\Pr_{\tilde{D}}[f'(R) = y] \geq cn/(2 \cdot 2^{0.5n \log(n)})$ and for every $y \in \text{Im}(f') \setminus \{\bot\}$. Moreover, for every $r$ such that $f'(r) = y$, we have
\[
\Pr[\tilde{D}] = \Pr[\tilde{D} | y] = \frac{\Pr_{\tilde{D}}[r \wedge y]}{\Pr_{\tilde{D}}[y]} \leq \frac{2^{-\left(\frac{5n}{6} \log \ell + n\right)}}{(en/2) \cdot 2^{-0.5n \log n}}
\]
\[
= 2^{-\left(\frac{5}{6} \log \ell -0.5 \log n + 1\right)n} \cdot (2/\epsilon n),
\]
which suggests $\tilde{D}$ has min-entropy at least $(\frac{5}{6} \log \ell - 0.5 \log n + 1) \cdot n - \log(2/\epsilon n)$.

We argue that the min-entropy of at least $n' = n/3$ blocks, conditioned on the outcome of all prior blocks as well as $y_i$, is at least $\log(n^{1.5})$. Towards a contradiction, assume otherwise. Let $V$ be the set of blocks with min-entropy at least $\log(n^{1.5})$ and let $W$ be the set of blocks with min-entropy less than $\log(n^{1.5})$ (as defined in Theorem 5). We will show that if $|V| \leq n/3$ there exists a point $r$ in the support of $\tilde{D}$ such that $\Pr_{\tilde{D}}[r] > 2^{-\left(\frac{5}{6} \log \ell -0.5 \log n + 1\right)n} \cdot (2/\epsilon n)$, which contradicts the min-entropy of $\tilde{D}$.

First, find any value $r^*_V$ such that $\Pr_{\tilde{D}}[R_V = r^*_V] \geq \frac{1}{\ell |V|}$. Note that $r^*_V$ must exist since the support size of $R_V$ is at most $\ell |V|$.

Let $S_W(r^*_V) = \{r : r_V = r^*_V \wedge \Pr_{\tilde{D}}[R = r] > 0\}$.

\[
\Pr_{\tilde{D}}[R \in S_W(r^*_V)] = \Pr_{\tilde{D}}[R_V = r^*_V] \geq \frac{1}{\ell |V|}.
\]

Second, we show that $|S_W(r^*_V)| \leq (2 \cdot n^{1.5}) |W|$. Consider any $r \in S_W(r^*_V)$. Applying the fourth condition of Theorem 5 with $\delta = 1$, condition on any $y \in \text{Im}(f') \setminus \{\bot\}$, for any $i \in W$ and any fixing of $R_{< i} = r_{< i}$, the number of elements in the support of $R_i | r_{< i}$ is at most $2 \cdot n^{1.5}$, which implies that $|S_W(r^*_V)|$ must be at most $(2 \cdot n^{1.5}) |W|$, since the positions for $V$ are fixed to $r^*_V$.

Based on the above two arguments, by the averaging argument, there must be some $r^* \in S_W(r^*_V)$ for which
\[
\Pr_{\tilde{D}}[R = r^*] \geq \frac{1}{\ell |V|} \cdot \frac{1}{(2 \cdot n^{1.5}) |W|}.
\]

50
Therefore, we have

\[- \log \Pr_{\delta} (r^*) = |V| \log(\ell) + |W| \log(2n^{1.5})\]

\[= |V| \log \ell + |W| + 1.5(n - |V|) \log n\]

\[\leq n + |V| \log(\ell/n) + 1.5n \log n\]

\[\leq n + n/3 \log(\ell/n) + 1.5n \log n\]

\[= n + n/3 \log(\ell) - 1/3 \log n + 1.5n \log n,\]

where the second to last line follows assuming \(|V| < n/3\).

To reach contradiction to (7), we require that

\[n + n/3 \log(\ell) - 1/3 \log n + 1.5n \log n \leq \left(\frac{8}{9} \log \ell - 0.5 \log n + 1\right) \cdot n - \log(2/\epsilon n).\]

The above is implied by

\[5/3n \log n \leq 5/9n \log \ell - \log(2/\epsilon n).\]

When \(\ell \geq 4n^3\) the above is implied by

\[5/3n \log n \leq 5/3n \log n + 10/3n - \log(2/\epsilon n),\]

which is true for \(n \geq \log(1/\epsilon) = \kappa\). Thus we reach contradiction to (7). We therefore conclude that \(|V| \geq n/3\).

**F.5 Proof of Lemma 9**

Note that we can equivalently view \(r'\) in the support of \(\hat{D}\) as a set of size \(n'\), or as a stream of elements of length \(n'\), where the element in the \(i\)-th block (for \(i \in [n']\)) comes from universe \(U_i\), and \(\{U_1, \ldots, U_{n'}\}\) are mutually disjoint. Taking the second view, given \(r', \bar{r}'\) in the support of \(\hat{D}\), we have that \(|r' \cap \bar{r}'| = z\) if and only if there exists some set \(Z \subseteq [n']\) of size \(z\) such that (1) the ordered set of elements in the blocks of \(r'\) indexed by \(Z\) (denoted \(r'_Z\)) is equal to the ordered set of elements in the blocks of \(\bar{r}'\) indexed by \(Z\) (denoted \(\bar{r}'_Z\)) and (2) the set of elements in the blocks of \(r'\) indexed by \([n'] \setminus Z\) (denoted \(r'_{\bar{Z}}\)) and the set of elements in the blocks of \(\bar{r}'\) indexed by \([n'] \setminus Z\) (denoted \(\bar{r}'_{\bar{Z}}\)) are disjoint.
We are now ready to analyze the probability that $|r' \cap \bar{r}'| = z$ for $r', \bar{r}'$ drawn from $\mathcal{D}$, and for $z \in [n']$:

$$
\Pr_{r', \bar{r}' \leftarrow \mathcal{D}}[|r' \cap \bar{r}'| = z] = \sum_{Z \subseteq [n'], |Z| = z} \Pr_{r', \bar{r}' \leftarrow \mathcal{D}}[(r'_Z = \bar{r}'_Z) \land (r'_Z \cap \bar{r}'_Z = \emptyset)]
$$

$$
\leq \sum_{Z \subseteq [n'], |Z| = z} \Pr_{r', \bar{r}' \leftarrow \mathcal{D}}[r'_Z = \bar{r}'_Z]
$$

$$
\leq \sum_{Z \subseteq [n'], |Z| = z} \left(\frac{1}{\sqrt{\frac{e}{3}n}}\right)^z
$$

$$
\leq \left(\frac{\sqrt{3}e}{z}\right)^z \cdot \left(\frac{1}{\sqrt{\frac{e}{3}n}}\right)^z
$$

$$
\leq \left(\frac{1}{\sqrt{n^{0.5}}}\right)^z.
$$