A New Sampling Protocol and Applications to Basing Cryptographic Primitives on the Hardness of **NP**

Iftach Haitner* and Mohammad Mahmoody† and David Xiao‡

January 5, 2010

Abstract

We investigate the question of what languages can be decided efficiently with the help of a recursive collision-finding oracle. Such an oracle can be used to break collision-resistant hash functions or, more generally, statistically hiding commitments. The oracle we consider, Sam_d where d is the recursion depth, is based on the identically-named oracle defined in the work of Haitner et al. (FOCS '07). Our main result is a constant-round public-coin protocol "AM–Sam" that allows an efficient verifier to emulate a Sam_d oracle for any constant depth $d = O(1)$ with the help of a $\mathbf{BPP}^{\mathbf{NP}}$ prover. AM–Sam allows us to conclude that if L is decidable by a k-adaptive randomized oracle algorithm with access to a $\mathsf{Sam}_{O(1)}$ oracle, then $L \in \mathbf{AM}[k] \cap \mathbf{coAM}[k]$.

The above yields the following corollary: assume there exists an $O(1)$-adaptive reduction that bases constant-round statistically hiding commitment on **NP**-hardness, then $\mathbf{NP} \subseteq \mathbf{coAM}$ and the polynomial hierarchy collapses. The same result holds for any primitive that can be broken by $\mathsf{Sam}_{O(1)}$ including collision-resistant hash functions and $O(1)$-round oblivious transfer where security holds statistically for one of the parties. We also obtain non-trivial (though weaker) consequences for k-adaptive reductions for any $k = \mathrm{poly}(n)$. Prior to our work, most results in this research direction either applied only to non-adaptive reductions (Bogdanov and Trevisan, SIAM J. of Comp. '06) or to primitives with special regularity properties (Brassard FOCS '79, Akavia et al., FOCS '06).

The main technical tool we use to prove the above is a new constant-round public-coin protocol (SampleWithSize) that we believe may be interesting in its own right, and that guarantees the following. Given an efficient function f on n bits, let D be the output distribution $D = f(U_n)$, then SampleWithSize allows an efficient verifier Arthur to use an all-powerful prover Merlin's help to sample a random $y \leftarrow D$ along with a good multiplicative approximation of the probability $p_y = \Pr_{y' \leftarrow D}[y' = y]$. The crucial feature of SampleWithSize is that it extends even to distributions of the form $D = f(U_{\mathcal{S}})$, where $U_{\mathcal{S}}$ is the uniform distribution on an efficiently decidable subset $\mathcal{S} \subseteq \{0,1\}^n$ (such D are called efficiently samplable with *post-selection*), as long as the verifier is also given a good approximation of the value $|\mathcal{S}|$.

Keywords: sampling protocols; collision-resistant hash functions; constant-round statistically hiding commitments; black-box lower bounds.

*Microsoft Research, New England Campus, iftach@microsoft.com.

†Princeton University, mohammad@cs.princeton.edu. Supported by NSF grants CNS-0627526, CCF-0426582 and CCF-0832797.

‡LRI, Université Paris-Sud, dxiao@lri.fr.

1 Introduction

The ability to sample from efficiently decidable sets (i.e., membership in such set can be decided efficiently, but sampling from the set might be hard) is an extremely powerful computation resource, to the point that having such ability for *any* decidable set implies $\mathbf{P} = \mathbf{NP}$. In this work we study less powerful samplers, that only agree to sample from more carefully chosen sets. We show that while these samplers can be used to break certain cryptographic primitives, they seem not to be strong enough to decide arbitrary **NP** languages. We then use this fact to give negative evidence on the possibility of basing such primitives on **NP** hardness.

Consider the sampler that gets a circuit C over $\{0,1\}^n$ as input, and outputs two random values x and x' in $\{0,1\}^n$ such that $C(x) = C(x')$. Such a sampler is known as a "collision finder", and breaks the security of any family of collision-resistant hash functions [61].[1] We consider the following generalization of the above sampler: the sampler Sam_d, where $d \in \mathbb{N}$, gets up to d recursive calls, each of the form $(C_1, \ldots, C_i, x)$, where $i \leq d$, each of the C_j's is a circuit over $\{0,1\}^n$ and $x \in \{0,1\}^n$. Sam_d answers depth 1 calls $(C_1, \cdot)$ with a random element in $\{0,1\}^n$. For depth $i > 1$ calls, Sam_d first checks that it was previously queried with $(C_1, \ldots, C_{i-1}, \cdot)$ and answered with x (otherwise, it aborts). If the check passes, then Sam_d answers with a random element in $C_1^{-1}(C_1(x)) \cap \ldots \cap C_i^{-1}(C_i(x))$. (See Section 2.1 for a more detailed description of Sam_d). (Note that the "collision finder" we described above is equivalent to Sam_2.) Such a sampler is very powerful, as it can be used for breaking the binding of any d-round statistically hiding commitments [63, 32].

Commitment schemes are the digital analogue of a sealed envelope. In such a scheme, a sender and a receiver run an interactive protocol where a sender commits to a bit b. In case the commitment is statistically hiding, then the protocol guarantees that from the receiver's point of view there exists roughly equal chance that the sender has committed to $b = 0$ or $b = 1$ (hence the bit b is hidden from the receiver information-theoretically). Where the additional guarantee is that a computationally-bounded sender can only find one way to decommit. (See Section 5.1.1 for the formal definition). Statistically hiding commitments are widely used throughout all of cryptography, with applications including, but not limited to, constructions of zero-knowledge protocols [13, 48, 23, 7, 33], authentication schemes [15], and other cryptographic protocols (e.g., coin-tossing [43]). Hence, it is highly important to study the minimal assumptions required for building them. Since Sam_d breaks any d-round statistically hiding commitments, it is very informative to learn what hardness assumptions Sam_d does *not* break (in particular, we have little hope to base d-round statistically hiding commitments on such assumptions). The following theorem shows that for constant d, Sam_d is not "too powerful".

Theorem 1.1 (Main theorem, informal)**.** *For any $d = O(1)$ and any efficient oracle-aided algorithm A, there exists a constant-round public-coin protocol* AM–Sam *with the following guarantee: either the output of the efficient verifier is statistically close to the output of A^{Sam_d}, or (if the prover cheats) the verifier aborts with high probability. Furthermore, the honest prover has complexity $\mathbf{BPP}^{\mathbf{NP}}$, while the cheating prover may be unbounded.*

We apply this theorem to understand what languages can be efficiently decided by randomized oracle-aided algorithms with oracle access to $\mathsf{Sam}_{O(1)}$, where the strength of the implication is a result of the adaptivity of the calls to $\mathsf{Sam}_{O(1)}$ made by the algorithm. We write $A \in \mathbf{BPP}^{\mathcal{O}[k]}$

[1] A family of collision resistant hash functions is a family of compressing functions with the following security guarantee: given a random function h in the family, it is hard to find $x \neq x'$ satisfying $h(x) = h(x')$.

to mean that R is a *k-adaptive* randomized oracle-aided algorithm using an oracle $\mathcal{O}$: A makes k adaptive rounds of queries to its oracle; each round may consist of many queries, but all of the queries in one round can be computed without looking at the oracle responses to any of the other queries in the same or later rounds. We say A is non-adaptive if $k = 1$.

We can apply Theorem 1.1 to obtain a k-round protocol for any language $L \in \mathbf{BPP}^{\mathsf{Sam}_{O(1)}[k]}$. Since $\mathbf{BPP}^{\mathsf{Sam}_{O(1)}[k]}$ is closed under complement, the above implies the following corollary.

Corollary 1.2 (Limits of languages decidable using oracle access to $\mathsf{Sam}_{O(1)}$)**.** *It holds that* $\mathbf{BPP}^{\mathsf{Sam}_{O(1)}[k]} \subseteq \mathbf{AM}[k] \cap \mathbf{coAM}[k]$. *In particular, every* $L \in \mathbf{BPP}^{\mathsf{Sam}_{O(1)}[k]}$ *has a k-round interactive proof where the honest prover has complexity* $\mathbf{BPP}^{\mathbf{NP}}$. *Furthermore, if L is* **NP***-complete, then the following consequences hold.*

$k = \mathrm{poly}(n)$**:** **co-NP** *has a public-coin $O(k)$-round interactive proof with honest prover complexity* $\mathbf{BPP}^{\mathbf{NP}}$.

$k = \mathrm{polylog}(n)$**:** *the subexponential hierarchy collapses to its third level (by [53]).*

$k = O(1)$**:** $\mathbf{PH} = \Sigma_2$ *(by [11]).*

Since the polynomial hierarchy is widely conjectured not to collapse, it follows that **NP**-complete languages are unlikely to be in $\mathbf{BPP}^{\mathsf{Sam}_{O(1)}[k=O(1)]}$. For $k = \mathrm{polylog}(n)$, the collapse is less understood, but it is still reasonable to conjecture that such a collapse does not occur. For $k = o(n)$ the consequence may not be implausible but would nevertheless lead to surprising progress on the long-standing open question of reducing the round complexity of interactive proofs for **co-NP** [44]. Finally for $k = \mathrm{poly}(n)$, as pointed out to us by Holenstein [38], it would answer a long-standing open question of Babai et al. [5] about reducing the complexity of the prover in interactive proofs for **co-NP** from $\mathbf{BPP}^{\#\mathbf{P}}$ to $\mathbf{BPP}^{\mathbf{NP}}$ (in fact this question is even open for multi-prover interactive proofs). Thus, depending on the adaptivity k, Corollary 1.2 gives an indication of either the implausibility or the difficulty of proving that **NP**-complete languages can be decided using the help of $\mathsf{Sam}_{O(1)}$.

1.1 Application to Basing Cryptography on NP-Hardness

Much of modern cryptography relies on computational intractability assumptions; starting with seminal works of Diffie and Hellman [18] and Goldwasser and Micali [29], the security of many if not most modern cryptosystems rests on the assumption that some underlying computational problem is hard to solve efficiently. Often the underlying problem is a concrete number-theoretic or algebraic problems [56, 19, 54]; unfortunately the existence of sub-exponential algorithms for factoring [14] and of efficient quantum factoring algorithms [60] have thrown into question whether many of these underlying assumptions are viable, and indeed faster factoring algorithms often translate into better attacks on the cryptosystems based on factoring. In light of this, there has been a search for more robust underlying intractability assumptions.

The holy grail of this search would be to base cryptography on the minimal assumption of $\mathbf{P} \neq \mathbf{NP}$; namely to show that $\mathbf{P} \neq \mathbf{NP}$ implies the existence of one-way functions, or, even more desirably, the existence of stronger cryptographic primitives such as collision-resistant hash functions or public-key cryptosystems. Other than the fact that $\mathbf{P} \neq \mathbf{NP}$ is necessary for the existence of one-way functions (and almost all other cryptographic primitives [39, 51]), the former is a "worst-case" assumption while the latter is of "average-case" nature, hence making the first

assumption is much more desirable. In fact, this goal dates back to the seminal paper by Diffie and Hellman [18].

Most constructions and proofs in the cryptographic literature are black-box, so it is worthwhile to understand whether black-box reductions can base cryptographic primitives on **NP**-hardness. A black-box reduction (also known as, black-box proof of security) from **NP**-hardness to the security of a cryptographic primitive is an efficient randomized oracle algorithm R such that given any oracle $\mathcal{O}$ that breaks the security of the cryptographic primitive, $R^{\mathcal{O}}$ solves SAT. The question of whether black-box reductions can be used to base cryptography on **NP**-hardness has previously been studied in [12, 20, 9, 22, 3, 52].

Since $\mathsf{Sam}_{O(1)}$ breaks a the security of d-round statically hiding commitments, it also breaks the wide variety of cryptographic primitives that yield such commitment via constant-adaptive black-box reductions. This list includes: collection of claw-free permutations with an efficiently-recognizable index set [23], collision-resistant hash functions [17, 47], (singly) homomorphic encryption [40], constant-round protocols for oblivious-transfer and private information retrieval schemes where the security of one of the parties holds information theoretically [32], the average-case hardness of **SZKP** [50], constant-round statistically *binding* commitments secure against selective opening attacks [64], and constant-round inaccessible entropy generators [34]. The following corollary states that if any of the above primitives can be based on **NP**-hardness via a black-box reduction R, then $R^{\mathsf{Sam}_{O(1)}}$ decides SAT.

Corollary 1.3 (immediate by Corollary 1.2)**.** *Let R be a k-adaptive reduction that bases the existence of any cryptographic primitive that can be broken by $\mathsf{Sam}_{O(1)}$ on **NP**-hardness. Then* $\mathrm{SAT} \subseteq \mathbf{AM}[k] \cap \mathbf{coAM}[k]$*, where the honest provers that realize this containment are in* $\mathbf{BPP}^{\mathbf{NP}}$*. The various consequences for different k given in Corollary 1.2 also hold.*

We remark that previous results studying the analogous question of basing (general) one-way functions on **NP**-hardness were restricted to non-adaptive reductions [20, 9, 3]. Other works do consider adaptive reductions, but with respect to more structured primitives [12, 3, 22]. See Section 1.3.1 for the description of previous works.

1.2 Main Tool — A New Sampling Protocol

Our main tool for proving Theorem 1.1, which is also our main technical contribution, is a new constant-round public-coin sampling protocol that we believe to be of independent interest. A distribution D is called *efficiently samplable* if it is the output distribution of an efficient function $f : \{0,1\}^n \to \{0,1\}^*$ (i.e., $D = f(U_n)$). A distribution is efficiently samplable *with post-selection* if $D = f(U_{\mathcal{S}})$ where $U_{\mathcal{S}}$ is the uniform distribution over an efficiently decidable set $\mathcal{S} \subseteq \{0,1\}^n$. Such distributions have previously been studied in the context of randomized algorithms [36]. We emphasize that although $\mathcal{S}$ is efficiently decidable, it is not necessarily possible to efficiently sample uniform elements of $\mathcal{S}$.

Our "Sample With Size" protocol takes $f, \mathcal{S}$, and a good approximation of $|\mathcal{S}|$ as input, and enables an efficient verifier to sample a uniform $y \in f(\mathcal{S})$, along with a good approximation of the value $p_y = \Pr_{y' \leftarrow f(U_{\mathcal{S}})}[y' = y]$.

Lemma 1.4. *(Sampling With Size protocol, informal) There exists a constant-round public-coin protocol* SampleWithSize*, where the parties get as a common input an efficiently decidable set $\mathcal{S} \subseteq$*

$\{0,1\}^n$, *an efficiently computable function* $f : \mathcal{S} \to \{0,1\}^*$ *and a good approximation (i.e., within* $(1 \pm \frac{1}{\mathrm{poly}(n)})$ *factor) of* $|\mathcal{S}|$ *and has the following guarantees:*

Either $\mathsf{V}_{\mathsf{SWS}}$ *outputs a pair* (x, s_x) *such that 1.* x *is distributed* $(1/\mathrm{poly}(n))$*-close to the uniform distribution over* $\mathcal{S}$*, and 2.* s_x *is a good approximation for* $|f^{-1}(f(x))|$ *, or (if the prover cheats) the verifier aborts with high probability. Furthermore, the honest prover has complexity* $\mathbf{BPP}^{\mathbf{NP}}$*, while the cheating prover may be unbounded.*

1.3 Related Work

1.3.1 NP-hardness and cryptography

Brassard [12] showed that if there exists a deterministic black-box reduction from **NP**-hardness to inverting a one-way permutation, then **NP**= **co-NP**. Bogdanov and Trevisan [9], building on earlier work of Feigenbaum and Fortnow [20], showed that if there exists a *non-adaptive* randomized black-box reduction from **NP**-hardness to inverting a one-way function (or more generally, to a hard on the average problem in **NP**), then $\mathbf{NP} \subseteq \mathbf{AM} \cap \mathbf{coAM}/\mathrm{poly}$, which is considered implausible since the polynomial hierarchy would collapse to the third level [65]. Akavia et al. [3] improved this result for the case of reductions to inverting one-way functions to show that the same hypothesis implies the uniform conclusion $\mathbf{NP} \subseteq \mathbf{AM} \cap \mathbf{coAM}$, which implies that the polynomial hierarchy collapses to the second level [11]. [3] also showed that if there exists an arbitrarily adaptive black-box reduction from **NP**-hardness to inverting one-way functions with *efficiently decidable range and efficiently computable preimage size* (decidable range, for short), then $\mathbf{co\text{-}NP} \subseteq \mathbf{AM} \cap \mathbf{coAM}$.[2] Goldreich and Goldwasser [22] showed that adaptive reductions basing public-key encryption schemes with the special property that the set of invalid ciphertexts is verifiable in **AM**, on **NP**-hardness would also imply that $\mathbf{NP} \subseteq \mathbf{coAM}$. Finally, Pass [52] takes a different route and showed that if a *specific type* of *witness-hiding* protocol exists, then an arbitrarily adaptive reduction from **NP**-hardness to the existence of one-way functions implies that $\mathbf{co\text{-}NP} \subseteq \mathbf{AM} \cap \mathbf{coAM}$. As recently point out by Haitner et al. [35], however, it is unlikely that known witness-hiding protocols are of the type required by [52].

We remark that while most cryptographic reductions we know of are non-adaptive, there are a few notable exceptions, in particular security reductions for building interactive protocols [48], pseudorandom number generators [37], and certain lattice-based cryptosystems [2, 45]. One may hope in particular that lattice problems might someday be used to prove that $\mathbf{P} \neq \mathbf{NP}$ implies one-way functions, since they already exhibit a worst-case to average-case hardness reduction.[3] [9, 3] do not rule out the possibility that any of these (or some other adaptive) techniques may succeed.

[2] As pointed out by [10], the proof of [3] indicates that decidable range one-way functions seem to be of different type than standard one way functions; their proof shows that it is unlikely to base *worst case* decidable range one-way functions on **NP**-hardness, where it is easy to show that worst case one-way functions *can* be based on **NP**-hardness. The above should be compare with the implications of Corollary 1.3, which also hold with respect to weak forms of statically hiding commitments (ones that only slightly hiding, or their binding cannot be broken with probability $1 - 1/\mathrm{poly}$). Such weak commitments, however, are not known to be implied by (standard) one way functions.

[3] In particular, the adaptivity of the lattice-based schemes seems essential for giving the best known approximation-ratio required in the worst-case hard lattice problem. Unfortunately, even in the best known reductions the starting worst-case hard problem in the $\mathbf{NP} \cap \mathbf{co\text{-}NP}$.

1.3.2 The oracle Sam

Simon [61] considered the sampler Sam_2^π — a generalization of Sam_2 that gets circuits with π-gates, where π is a random permutation oracle. He showed that while Sam_2^π breaks any collision-resistant hash functions relative to random permutation π (i.e., the hash function is allowed to use π-gates), it cannot invert π. Continuing this line research, Haitner et al. [32] showed that Sam_d^π breaks all d-round statistically hiding commitments, even those implemented using π, but Sam_d^π does not help to invert π if $d = o(n/\log n)$. As a consequence, the above results rule out the possibility of basing $o(n/\log n)$-round statistically hiding commitments on the existence of one-way functions/permutations, using *fully-black-box* reductions — a reduction from statistically hiding commitments to one-way function is fully-black-box, if the proof of security is black box (in the sense of Corollary 1.3), where in addition the construction uses the one-way function as black-box. Note that these results are incomparable to the result stated in Corollary 1.3. On one hand, they rule out all fully-black-box reductions *unconditionally* without restrictions on adaptivity. On the other hand, they consider reductions starting from one-way functions rather than **NP**-hardness, and their results do not apply to non-black-box constructions. In contrast, Corollary 1.3 also applies to reductions where the construction is non-black-box, which permits, for example, the construction to exploit the fact that YES instances of **NP** languages have efficiently verifiable witnesses. In other words, Corollary 1.3 only requires that the *security analysis* be black-box. We refer the reader to Reingold et al. [55], which, although they do not focus on our case where the construction is non-black-box but the security analysis is black-box, nevertheless is useful for understanding the distinctions between various notions of reductions.

Sam and zero knowledge. In recent work, Gordon et al. [31] observe that our main result is useful in the context of understanding zero knowledge proofs. In particular, they prove using Theorem 1.1 that if a language L has a constant-round black-box computational zero-knowledge proof based on one-way permutations with a k-adaptive simulator, then $L \in \mathbf{AM}[k] \cap \mathbf{coAM}[k]$. Their result suggests that reducing the round complexity of known constructions of zero knowledge proofs based on one-way permutations for **NP** (e.g., [26, 8]) (all of which have super-constant round complexity) to a constant number of rounds is implausible if the simulator must also be $O(1)$-adaptive or at least very difficult to prove for larger adaptivity.

1.3.3 Efficiently decidable sets

Estimating statistics. Estimating statistical properties of efficiently samplable distributions has long been studied in theoretical computer science [30, 21, 1, 49, 58, 24]. Typically, estimating interesting parameters of samplable distributions (and therefore also of samplable distributions with post-selection) such as entropy or statistical difference is hard (e.g., **SZKP**-hard). Nevertheless, for samplable distributions it was known that an efficient verifier can estimate various parameters in constant rounds with the help of an all-powerful prover.

Bounding set-size protocols. The constant-round public-coin lower bound protocol of Goldwasser and Sipser [30] (see Lemmas 3.16 and 3.17 for the formal statements) can be used to lower-bound the size of efficiently decidable sets. Namely, on input an efficiently decidable set $\mathcal{S}$ and a value s, the prover makes the verifier accept iff $|\mathcal{S}| \geq s$. Fortnow [21] (see also Aiello and Håstad [1]) gives a constant-round protocol that upper-bounds the sizes of efficiently decidable sets

$\mathcal{S}$ where *in addition* the verifier has a uniform element of $\mathcal{S}$ that is *unknown* to the prover (see Lemma B.1).

These protocols are related to our protocol SampleWithSize. For example, one can estimate with respect to $D = f(U_n)$ the probability $p_y = \Pr_{y' \leftarrow D}[y' = y]$ for a random $y \leftarrow D$ by lower-bounding and upper-bounding the set $|f^{-1}(y)|$. In particular, the upper bound [21, 1] can be applied in this case, since the verifier can sample $x \leftarrow U_n$, compute $y = f(x)$ and ask the prover for an upper bound on the size of the set $f^{-1}(y)$ without revealing x. This is one way to prove SampleWithSize for the special case $\mathcal{S} = \{0,1\}^n$.

We cannot necessarily apply, however, the upper bounds of [21, 1] to do the same thing with *post-selected* distributions $D = f(U_{\mathcal{S}})$. That is, even though $f^{-1}(y)$ is efficiently decidable, it may not be possible to efficiently generate $x \leftarrow U_{\mathcal{S}}$ conditioned on $x \in f^{-1}(y)$ (with x that is hidden from the prover). As we will discuss in Section 2.1.1, handling post-selected distributions is necessary to obtain Theorem 1.1. Although one-sided lower-bound estimates can be obtained from the lower-bound protocol of [30], it is unknown how to get two-sided estimates using the upper bound protocol of [21, 1], where the difficulty is to obtain secret samples from $U_{\mathcal{S}}$. In contrast, SampleWithSize *does* guarantee a two-sided bound for the estimate $p_y \approx \Pr_{y' \leftarrow D}[y = y']$ for a random $y \leftarrow f(U_{\mathcal{S}})$.

Sampling. Using an all-powerful prover to help sample is an old question in computer science, dating at least to the works of Valiant and Vazirani [62] and Impagliazzo and Luby [39]. In building SampleWithSize, we will use a sampling protocol that first appeared in Goldreich et al. [28], and was further refined by Akavia et al. [3]. This constant-round public-coin protocol takes as input an efficiently decidable set $\mathcal{S}$ and a good approximation of $|\mathcal{S}|$, and outputs a nearly-uniform element of $\mathcal{S}$. See Lemma 3.18 for a formal statement. Our protocol SampleWithSize uses their sampling protocol and extends it by also giving set size information about the sample that is generated.

Another protocol that seems related to SampleWithSize is the random selection protocol of Goldreich et al. [25]. Their protocol accomplishes a goal similar to the protocol of [28], allowing a verifier to select a random element of a set. Their protocol, however, cannot be applied in our context as it requires super-constant round complexity. Other related work include random selection protocols arising in the study of zero knowledge [16, 27, 59], but none of these protocols provides the size information that is provided by SampleWithSize).

1.4 $\mathsf{Sam}_{O(1)}$ Vs. Sam_2

It is worthwhile noting that Theorem 1.1 for non-recursive collision finders (i.e., Sam_2), can be proven via a straightforward application of the lower-bound protocol of Goldwasser and Sipser [30] (see Lemma 3.17) and the upper-bound protocol of [21, 1] (see Lemma B.1). See Appendix B for an illustration of these easier proofs.

Various evidence suggests, however, that $\mathsf{Sam}_{O(1)}$ is more powerful than Sam_2. There is no known way to "collapse" the depth (i.e., to show that Sam_2 suffices to emulate Sam_d for $d > 2$), and under various assumptions there exist problems solvable using $\mathsf{Sam}_{O(1)}$ but not Sam_2 (for example the average-case hardness of **SZKP** [50] and constant-round parallelizable zero knowledge proofs for **NP** [34], which both imply constant-round statistically hiding commitment, but not collision-resistant hash functions). Therefore, we do not focus on the (admittedly simpler) proof of Main Theorem 1.1 for the case of Sam_2, and rather we build our machinery SampleWithSize in order to prove Theorem 1.1 for the case of $\mathsf{Sam}_{O(1)}$.

Organization

High level description of our techniques is given in Section 2. Notations, definitions and basic lemmas can be found in Section 3. Our main technical contribution (Lemma 4.1) is given in Section 4, where our main result (Theorem 5.2) and its applications to understanding black-box reductions basing cryptography on **NP**-hardness are given in Section 5. Finally, in Appendix B we sketch a simple proof of Theorem 5.2 for the case of $d = 2$.

2 Our Techniques

In this section we overview the main ingredients used for proving Theorem 1.1. We first show how to use protocol SampleWithSize (i.e., the new sampling protocol) to implement protocol AM–Sam, and then give details on the implementation of protocol SampleWithSize itself.

2.1 Using SampleWithSize to implement AM–Sam

We start with more precise description of Sam_d (the algorithm that AM–Sam tries to "emulate"). On input $(C_1, \ldots, C_i, x)$, where $x \in \{0,1\}^n$ and each C_j is a circuit over $\{0,1\}^n$, Sam_d performs the following "sanity check": it checks that $i \leq d$, and if $i > 1$ then it also checks that it was previously queried on $(C_1, \ldots, C_{i-1}, x')$ (for some $x' \in \{0,1\}^n$) and answered with x. If any of these checks fail, Sam_d returns "failure". Otherwise, Sam_d returns a random element x' in $\mathcal{S}(C_1, \ldots, C_{i-1}, x) := \{x' \in \{0,1\}^n : \forall 1 \leq j \leq i-1, C_j(x') = C_j(x)\}$ (if $i = 1$, it returns a random $x' \in \{0,1\}^n$). Viewed differently, x' is a random collision with x for depth with respect to $C_1, \ldots, C_{i-1}$ since it satisfies $C_j(x') = C_j(x)$ for every $1 \leq j \leq i-1$.

In protocol AM–Sam, the verifier chooses A's random coins at random and then emulates A^{Sam_d}, while answering each query $(C_1, \ldots, C_i, x)$ to Sam_d using the following subprotocol: The verifier first performs (using the data stored during previous executions, see below) the sanity check of Sam_d, and returns "failure" in case this tests fails. Otherwise it does the following:

In case $i = 1$: The verifier sets $\mathcal{S} = \{0,1\}^n$, $s = 2^n$, and $f = C_1$ and runs SampleWithSize to get a random sample $x_1 \in \{0,1\}^n$ and an approximation $s_1 \approx |\{x' \in \{0,1\}^n : C_1(x_1) = C_1(x')\}|$. The verifier stores an entry $((C_1, x_1), s_1)$ in its memory, and returns x_1 to A as the query's answer.

In case $i > 1$: The verifier looks up the entry $((C_1, \ldots, C_{i-1}, x), s_{i-1})$ from its memory (the sanity checks guarantee that such entry must exist, since x was the answer for a previous query $(C_1, \ldots, C_{i-1}, \cdot)$). Run SampleWithSize on $\mathcal{S} = \mathcal{S}_i = \{x' \in \{0,1\}^n : \forall 1 \leq j \leq i-1,\ C_j(x') = C_j(x)\}$, $f = C_i$, and s_{i-1} in order to obtain $x_i \in \mathcal{S}$ and the approximation $s_i \approx |\{x' \in \mathcal{S} : C_i(x_i) = C_i(x')\}|$. As in the case $i = 1$, the verifier stores an entry $((C_1, \ldots, C_i, x_i), s_i)$ in its memory, and returns x_i.

To see that AM–Sam indeed behaves like A^{Sam_d}, we first note that Lemma 1.4 yields that for depth 1 queries, SampleWithSize returns x_1 that is (close to) uniform in $\{0,1\}^n$, exactly as Sam_d. In addition, SampleWithSize outputs a good approximation $s_1 \approx |C_1^{-1}(C_1(x_1))|$ that can be used as input for depth 2 queries to AM–Sam. Since s_1 is a good approximation, this means that a depth 2 query will be answered by SampleWithSize with x_2 where x_2 is a near-uniform element

of $C_1^{-1}(C_1(x_1))$, just as Sam_d would answer. SampleWithSize also outputs a good approximation $s_2 \approx |C_2^{-1}(C(x_2))|$, which can be used for depth 3 queries, and so on.

The above is done in parallel for each of the k adaptive rounds of oracle queries. The approximation error of s_i grows as the depth increases, and from the formal statement of Lemma 1.4 (see Lemma 4.1) it follows that we can repeat the above process a constant number of times. Unfortunately, the accumulated error becomes super-polynomial for any $d = \omega(1)$.

Contrast to previous work [20, 9, 3]. Sam_d is in a sense a "canonical recursive collision finder". Similarly, one could consider a "canonical function inverter" that takes as input a circuit C and a value y and outputs a random element of $C^{-1}(y)$. Such an oracle would break all one-way functions. One could then ask whether it is possible to construct some kind of "AM-Inv" that emulates this canonical inverter. Such a result would strengthen our main theorem, since an inverter can in particular find collisions.

Unfortunately, it is not known how to build such an AM-Inv. The difficulty is handling cheating provers, who claim that the given query is not invertible. Notice that for the problem of inverting a function, it is possible to ask queries (C, y) where y is not in the image of C. In this case the oracle must say that the query is invalid. Since there is no efficient way to verify that y is *not* in the image of C, a cheating prover can claim, say, that none of the verifier's queries are in the image of C even when some are valid queries. In general, it is not known how to catch this kind of cheating, since proving that y is not in the image of C is a **co-NP** statement.

As already mentioned in Section 1.3, various works have gotten around this difficulty using additional restrictions either on the way the inverting oracle is called (i.e., non-adaptivity) or on the kinds of functions that the oracle inverts (i.e., functions with efficiently verifiable range and efficiently computable preimage sizes). The main reason we are able to build AM–Sam whereas building "AM-Inv" seems out of reach is that in our setting, unlike the inverting oracle, Sam_d can never respond "failure" to a well-formed query (one that passes the sanity checks), since the sanity check ensures that collisions always exist.

2.1.1 Necessity of post-selection

It seems necessary to have a good hold on the statistics of an efficiently samplable distributions with post-selection, in order to make AM–Sam successfully emulate Sam_d for any $d > 2$. This is clear from the sketch of how we use SampleWithSize to prove AM–Sam given in Section 2.1: in order for the proof to work for large depths, we need SampleWithSize to apply not just on the $\mathcal{S} = \{0,1\}^n$ but, on the sets $\mathcal{S}_i$ that are returned by previous calls to SampleWithSize. These sets $\mathcal{S}_i$ are efficiently decidable, but it may not be possible for the verifier to generate secret elements from $\mathcal{S}_i$. This prevents us from giving a simple proof of SampleWithSize using the upper bound protocol of [21, 1], since their protocol relies on being able to generate secret elements in $\mathcal{S}_i$.

2.2 Proving SampleWithSize

Approximating histograms. Underlying the proof of SampleWithSize is a constant-round public-coin protocol VerifyHist for verifying histograms of distributions. Let D be a distribution over $\{0,1\}^*$, and let $p_y = \Pr_{y' \leftarrow D}[y' = y]$. The histogram of D is a function $h : [0,1] \to [0,1]$, where $h(p) = \Pr_{y \leftarrow D}[p_y = p]$. Namely, the histogram tells us the distribution of weights of elements drawn

from D. Since we would like to consider objects with small description, our formal definition is a "discretized" variant of the above notion (see Definition 3.2 for a precise definition).

In this paper, we use the 1st Wasserstein distance $\mathsf{W1}$ (also known as Kantorovich distance and Earth Mover's distance, see Definition 3.8) as the distance measure between histograms. This distance is well studied in probability theory [46, 41, 42] and also has application in computer science, for example in the realm of image processing [57]. To understand this distance intuitively, think of a histogram h as piles of "earth" on the unit interval $[0,1]$, where the larger $h(p)$ is, the larger the pile at location $p \in [0,1]$. $\mathsf{W1}(h,h')$ is the *minimal amount of work* that must be done to push the configuration of earth given by h to get the configuration given by h'. Recall that in physics, work is equal to force times distance. So, for example, pushing a probability mass of 0.1 from position 1/10 to position 9/10 requires work $0.1 \cdot 8/10 = 8/100$, while pushing the same mass from 1/3 to 2/3 requires less work, only $0.1 \cdot 1/3 = 1/30$.

More formally, the $\mathsf{W1}$ distance for distributions over $[0,1]$ is defined as:

$$\mathsf{W1}(h,h') = \int_0^1 \left| \int_0^x (h(p) - h'(p))\, dp \right| dx.$$

The intuition is that $\int_0^x (h(p) - h'(p))\, dp$ captures the amount of mass "pushed" from the interval $[0,x]$ into the interval $(x,1]$, and taking an integral over all these amounts together gives us the total amount moved. As a comparison, it is easy to observe that $\mathsf{W1}(h,h') \leq \|h - h'\|_1$. We will define and use a suitably discretized version of this $\mathsf{W1}$ distance in Definition 3.8.

The protocol $\mathsf{VerifyHist}$ allows us to verify the validity of an histogram in the $\mathsf{W1}$ distance not just of efficiently samplable distributions, but even efficiently samplable distributions with *post-selection* (as defined in the previous section about $\mathsf{SampleWithSize}$), as long as the verifier knows roughly the probability of the post-selected event. Our protocol is as follows.

Lemma 2.1 (Verify Histogram Protocol, informal)**.** *There exists a constant-round public-coin protocol* $\mathsf{VerifyHist}$*, between a prover* $\mathsf{P_{VH}}$ *in* $\mathbf{BPP^{NP}}$ *and an efficient verifier* $\mathsf{V_{VH}}$*, where the parties get as a common input an efficiently decidable set* $\mathcal{S} \subseteq \{0,1\}^n$*, an efficiently computable function* $f : \mathcal{S} \to \{0,1\}^*$*, a good approximation (i.e., within* $(1 \pm \frac{1}{\mathrm{poly}(n)})$ *factor) of* $|\mathcal{S}|$ *and a claimed histogram* h *of the distribution* $D = f(U_{\mathcal{S}})$*, and has the following guarantees:*

Completeness. *If* h *is the histogram of* D*, then* $\mathsf{P_{VH}}$ *makes* $\mathsf{V_{VH}}$ *accept with high probability.*

Soundness. *If* h *is far from the histogram of* D *in the 1st Wasserstein distance, then no (possibly cheating) prover can make* $\mathsf{V_{VH}}$ *accept with non-negligible probability.*

Previous work using histograms. Previous works have used histograms to estimate set sizes, and a related protocol to $\mathsf{VerifyHist}$ appears in Goldreich et al. [28]. We emphasize that their protocol accomplishes a different task that is incomparable to ours.[4]

[4]The protocol of [28] lower bounds the size of a set $\mathcal{S}$ that is efficiently verifiable via a low-communication *interactive protocol*, but not efficiently decidable using a circuit. To do so, they recursively refining the histogram such that the following holds: if the prover lies about $|\mathcal{S}|$ (and gives an over-estimate), then at the base of the recursion the verifier catches the cheating by noticing that some parts of the histogram are empty. The prover, however, must claim they are non-empty in order to be consistent with previous answers.

2.2.1 Proving soundness of SampleWithSize using VerifyHist.

Since completeness is straightforward to observe from the definition of the protocols, in the following we focus on describing how to prove the soundness properties of the SampleWithSize and VerifyHist protocols. The overviews presented here assume familiarity with the lower bound protocol of [30] and the uniform sampling lemma of [28, 3]. An informal description of these protocols appears in Section 1.3.3, while formal statements appear in Section 3.6.

The verifier in SampleWithSize essentially does the following. Given $\mathcal{S}, f$ and s where $s \approx |\mathcal{S}|$, consider the post-selected distribution $D = f(U_{\mathcal{S}})$. Ask the prover for D's histogram h and run VerifyHist to verify that h is indeed close to the true histogram of D. Now:

1. Use the sampling protocol of [28] to sample many points: $x_1, \ldots, x_k \leftarrow U_{\mathcal{S}}$. Set $y_i = f(x_i)$.
2. The prover sends $s_1, \ldots, s_k$, which should be equal to $|f^{-1}(y_i)|$. Run the [30] lower bound to ensure that indeed $|f^{-1}(y_i)| \geq s_i$ for all i.
3. Use $s_1, \ldots, s_k$ to construct from this distribution an "empirical histogram" h^{emp} where each entry is computed this way: $h^{\mathsf{emp}}(p) = \frac{1}{k}|\{i \mid p = \frac{|s_i|}{|\mathcal{S}|}\}|$.[5].
4. If h^{emp} and h are too far apart in the 1st Wasserstein distance, then abort. Otherwise, pick a random $i \in [k]$ and output (x_i, s_i).

Intuitively, the lower bound in Step 2 means that if the prover wants to cheat, it can only claim s_i to be smaller than $|f^{-1}(y_i)|$. On the other hand, by Chernoff we know that for large enough k, the true empirical histogram of the examples $y_1, \ldots, y_k$ will be close to h. Therefore, if the prover consistently under-estimates $|f^{-1}(y_i)|$ for many i, then this implies he gives us a very distorted empirical histogram, and we will catch him in the check on the 1st Wasserstein distance in Step 4. Together, this implies that $s_i \approx |f^{-1}(y_i)|$ for almost all i, and so outputting (x_i, s_i) for a random i is good with high probability.

2.2.2 Proving soundness of VerifyHist.

Let us consider the very special case of VerifyHist where the input is of the form $\mathcal{S} = \{0,1\}^n$, $s = 2^n$, and f that is promised to be a k-to-1 regular function, but where the regularity k is unknown. This implies that $|f(\{0,1\}^n)| = 2^n/k$, and the only nonzero element of the histogram is $h(k/2^n)$, which has value 1. To verify a claimed value k', the verifier does the following.

Preimage test: The verifier runs the lower-bound protocol (Lemma 3.17) to check that $k = |f^{-1}(f(x))| \geq k'$ (here x is arbitrary and unimportant since the function is regular).

Image test: The verifier runs the lower-bound protocol to check that $2^n/k = |f(\{0,1\}^n)| \geq 2^n/k'$.

It is clear from the guarantee of the lower-bound protocol that the preimage test prevents the prover from claiming $k' \gg k$, and also that the image test prevents the prover from claiming $k' \ll k$, as this would make $|f(\{0,1\}^n)| = 2^n/k \ll 2^n/k'$ and the lower-bound of the image test would fail. We are able to use the lower-bound protocol in the image test because membership in the image of f can be efficiently verified.

[5]Here we ignore the fact that the proper way to construct the empirical histogram requires discretizing the probability space so that elements y, y' with $|f^{-1}(y)| \approx |f^{-1}(y)|$ contribute to the *same* entry in the histogram. See Definition 3.2

The general case. The idea in the above simple case is that by giving a lower bound on the image of the function f, one obtains an upper bound on the preimage. This idea extends to f that are far from being regular, and we generalize the special case of regular functions to a protocol with more image tests over many *subsets* of $f(\mathcal{S})$.

To be more precise, define the sets $\mathcal{W}_k \subseteq f(\mathcal{S})$ given by $\mathcal{W}_k = \{y \colon y \in f(\mathcal{S}), |f^{-1}(y)| \geq k\}$. Just as in the simple example of regular functions where knowing the regularity k tells us the size of the image $|f(\{0,1\}^n)|$, so for the general case we will observe that knowing the histogram of D tells us the sizes of $|\mathcal{W}_k|$. Let w_k^h be the estimate of $|\mathcal{W}_k|$ that is given by the histogram h. (Here we do not specify further how exactly this estimate is calculated, since it requires additional parameters and the formula is complicated. See Section 4 for details.)

Also, one can efficiently verify membership in $\mathcal{W}_k$, by using the lower-bound protocol: given y, run the lower-bound to verify that $|f^{-1}(y)| \geq k$. Therefore, the set sizes $|\mathcal{W}_k|$ can themselves be lower-bounded using (a generalization of) the lower-bound protocol of [30], given in Lemma 3.17.

Our VerifyHist protocol consists essentially of the following. Given an input $\mathcal{S}, f, s$ which defines the distribution $D = f(U_{\mathcal{S}})$ and a claimed histogram h for D, the verifier does the following

Preimage test: First, use the Sampling Lemma 3.18 to sample $y_1, \ldots, y_k \leftarrow D$. The prover also sends claimed sizes $s_1, \ldots, s_k$. The verifier constructs the empirical histogram h^{emp} given these claimed sizes and aborts if $\mathsf{W1}(h, h^{\mathsf{emp}})$ is too large. The verifier runs the lower-bound protocol (Lemma 3.17) to check that $|f^{-1}(y_i)| \geq s_i$ for all i and aborts if any lower-bound fails.

Image test: For $k = 2^{i\varepsilon}$ where i ranges over $1, \ldots, n/\varepsilon$ and $\varepsilon = 1/\operatorname{poly}(n)$ is a suitably small function, run the lower bound protocol to verify that $|\mathcal{W}_k| \geq w_k^h$.

The main technical contribution of our work is to prove that if all these tests accept, then h must be close to the true histogram in the 1st Wasserstein distance. The intuition is the same as for the special case of regular functions explained above: the preimage tests prevent the prover from claiming that preimages are larger than they actually are, while the image tests prevent the prover from claiming that many preimages are smaller than they actually are, because, as we prove in Section 4, under-estimating the size of many preimages distorts the histogram in such a way that there exists k where $w_k^h \gg |\mathcal{W}_k|$, and this will be caught by the image lower-bound protocol.

3 Preliminaries

3.1 Notation

We use calligraphic letters to denote sets and capital letters to denote random variables. Given $u \in \mathcal{U}^{m+1}$, we denote the components of u as $u = (u_0, \ldots, u_m)$, and let $u_{\leq i} = (u_0, \ldots, u_i)$. For a random variable X, we write $x \leftarrow X$ to indicate that x is selected according to X. Similarly, we write $x \leftarrow \mathcal{S}$ to indicate that x is selected according to the uniform distribution over the set $\mathcal{S}$. By $U_{\mathcal{S}}$ we denote the random variable whose distribution is uniform over $\mathcal{S}$. All the logarithms written as log in this paper are in base 2 . For any $m \in \mathbb{N}$, we let $[m] = \{1, \ldots, m\}$ and $(m) = \{0, 1, \ldots, m\}$.

Given a two-party protocol (A, B), we let the random variable $\langle \mathsf{A}, \mathsf{B} \rangle(x)$ denote (the transcript of) a random execution of the protocol over a common input x.

3.2 Many-wise Independent Functions

Definition 3.1 (d-wise independent functions)**.** *A family of functions $\mathcal{F}_{n,t} = \{f : \{0,1\}^n \to [t]\}$ is d-wise* independent *if for any $x_1, \ldots, x_d \in \{0,1\}^n$ and $y_1, \ldots, y_d \in [t]$, $\Pr_{f \leftarrow \mathcal{F}_{n,t}}[f(x_1) = y_1, \ldots, f(x_d) = y_d] = t^{-d}$.*

It is possible to efficiently sample d-wise independent functions for any $d = \text{poly}(n)$, for example by using a random univariate degree $d-1$ polynomial (for better parameters see for example [4]).

3.3 The Histogram of a Function

For any distribution D, let $p_y = \Pr_{y' \leftarrow D}[y = y']$ be the weight of an element y under the distribution D. The histogram of D is the probability distribution of weights of elements drawn from D. Namely, a histogram h assigns to every $p \in [0,1]$ the probability $h(p) = \Pr_{y \leftarrow D}[p = p_y]$. We will discretize the histogram on the log-scale with an approximation error ε to obtain the following definition.

Definition 3.2 (Histogram)**.** *Let $\mathcal{S} \subseteq \{0,1\}^n$, let f be a function from $\mathcal{S}$ to $\{0,1\}^*$, let $\varepsilon > 0$ and let $m = \lfloor n/\varepsilon \rfloor$. For $i \in (m)$ we define the i'th "bin" as $\mathcal{B}_i = \left\{y \colon \Pr_{x \leftarrow \mathcal{S}}[f(x) = y] \in (2^{-(i+1)\cdot\varepsilon}, 2^{-i\varepsilon}]\right\}$. Let $\mathsf{Bin}(x) := i$ iff $f(x) \in \mathcal{B}_i$, and let $h = (h_0, \ldots, h_m)$ where $h_i = \Pr_{x \leftarrow \mathcal{S}}[\mathsf{Bin}(x) = i]$. We call h the ε-*histogram of the function f over $\mathcal{S}$.

For simplicity, in the definition of a histogram, we always assume that ε is chosen in a way that $n/\varepsilon \in \mathbb{N}$ and $m = n/\varepsilon$ exactly. Notice that the bins with smaller numbers contain "heavier" elements (namely for smaller i, the elements $y \in \mathcal{B}_i$ occur with larger probability). The histogram h encodes the (approximate) regularity structure of the function f over the domain $\mathcal{S}$. For example let $\varepsilon = 1$ (which implies $m = n$) and $\mathcal{S} = \{0,1\}^n$. Therefore a 1-to-1 function's histogram has one non-zero entry $h_m = 1$, a 2-to-1 function's histogram has one non-zero entry $h_{m-1} = 1$, while a constant function's histogram has one non-zero entry $h_0 = 1$. Functions with more complex regularity structures would have more complex histograms.

Histograms can also be defined for *empirical* samples drawn according to a distribution as follows. Suppose we have a set of examples $x_1, \ldots, x_\ell$ all sampled from $x_i \leftarrow \mathcal{S}$, and suppose someone claims to us the weights of each of the $f(x_i)$. Namely, he gives to us a labeling function $v : [\ell] \to (m)$ with the claim that $f(x_i) \in \mathcal{B}_{v(i)}$. This labeling v induces a claimed histogram as follows.

Definition 3.3 (Empirical histogram)**.** *For a labeling function $v : [\ell] \to (m)$, define $h^v = \mathsf{Hist}(v)$ where $h^v_j = \Pr_{i \leftarrow [\ell]}[v(i) = j]$.*

The following observations easily follow from Definition 3.2.

Proposition 3.4.

1. *$\bigcup_{i \in (m)} \mathcal{B}_i = \mathcal{S}$ and $\sum_{i \in (m)} h_i = 1$,*
2. *For every $y \in \mathcal{B}_i$ it holds that $|f^{-1}(y)| \in (|\mathcal{S}| \cdot 2^{-(i+1)\varepsilon}, |\mathcal{S}| \cdot 2^{i\cdot\varepsilon}]$.*
3. *For every $i \in (m)$ it holds that $|\mathcal{B}_i| \in [h_i \cdot 2^{i\varepsilon}, h_i \cdot 2^{(i+1)\cdot\varepsilon})$.*

3.4 Metrics over Distributions

3.4.1 Statistical distance

Definition 3.5 (Statistical distance)**.** *Given two random variables X, Y over a common universe $\mathcal{U}$ let $\Delta(X,Y)$ denote their* statistical distance*, where $\Delta(X,Y) = \max_{\mathcal{S}\subseteq\mathcal{U}} |\Pr[X \in \mathcal{S}] - \Pr[Y \in \mathcal{S}]|$. X and Y are called δ-close iff $\Delta(X,Y) \leq \delta$.*

Two ensembles of distributions $\{X_n\}, \{Y_n\}$ over $\{0,1\}^{\mathrm{poly}(n)}$ are *statistically indistinguishable* if $\Delta(X_n, Y_n) \leq n^{-\omega(1)}$.

We will use the following elementary properties of the Statistical distance.

Lemma 3.6. *Let X be a random variable distributed over the set $(\mathcal{S}\times\mathcal{T})\cup\{\perp\}$, and suppose $\Delta(X, U_{\mathcal{S}\times\mathcal{T}}) \leq \Pr[X=\perp] + \delta$. Let $X_{\mathcal{S}}$ be the random variable that equals $\perp$ if $X = \perp$ and equals to $s \in \mathcal{S}$ if $X = (s,t)$. Then it holds that $\Delta(X_{\mathcal{S}}, U_{\mathcal{S}}) \leq \Pr[X=\perp] + \delta$.*

Lemma 3.7. *Let X, Y be a random variables distributed over the set $\mathcal{S}\cup\{\perp\}$ such that $\Pr[Y = \perp] = 0$ and $\Delta(X,Y) \leq \Pr[X=\perp] + \delta$. Then for any event $T \subset \mathcal{S}$ it holds that:*

$$\Pr[X \in T] = \Pr_{x\leftarrow X}[x \neq \perp \wedge x \in T] \leq \Pr[T] + \delta.$$

3.4.2 Wasserstein distance

The following metric measures how much "work" it takes to turn one distribution over $\mathcal{U}$ into another one; where the amount of work is assumed to be amount of needed "moves" of the probability masses times the distance by which they are moved. Our definition is for the special case that the members of $\mathcal{U}$ form a one-dimensional array and their distance is the normalized difference between their indexes. For more general spaces it is known as the 1st Wasserstein distance or the Kantorovich distance [46, 41, 42]. Also in the field of image processing it known as the Earth Mover's distance [57].

Definition 3.8 (1st Wasserstein distance over arrays)**.** *Given two distribution vectors x and y over (m), for every $i \in (m)$ we let $a_i = \sum_{j\in(i)} x_j$ and $b_i = \sum_{j\in(i)} y_j$. We let*

- $\overrightarrow{\mathsf{W1}}(x,y) = \frac{1}{m} \cdot \sum_{i\in(m):\ a_i > b_i} (a_i - b_i)$,
- $\overleftarrow{\mathsf{W1}}(x,y) = \frac{1}{m} \cdot \sum_{i\in(m):\ b_i > a_i} (b_i - a_i)$,
- $\mathsf{W1}(x,y) = \overrightarrow{\mathsf{W1}}(x,y) + \overleftarrow{\mathsf{W1}}(x,y)$,

where we call $\mathsf{W1}(x,y)$ the 1st Wasserstein distance *between x and y, where $\overleftarrow{\mathsf{W1}}(x,y)$ and $\overrightarrow{\mathsf{W1}}(x,y)$ are called the left and right Wasserstein distance respectively.*

The following proposition is easy to verify.

Proposition 3.9. *Let x, y and z be distributions vector over (m), then*

1. $\overrightarrow{\mathsf{W1}}(x,y) + \overrightarrow{\mathsf{W1}}(y,z) \geq \overrightarrow{\mathsf{W1}}(x,z)$,
2. $\overleftarrow{\mathsf{W1}}(x,y) + \overleftarrow{\mathsf{W1}}(y,z) \geq \overleftarrow{\mathsf{W1}}(x,z)$,

3. $\mathsf{W1}(x,y) + \mathsf{W1}(y,z) \geq \mathsf{W1}(x,z)$.

Proof Sketch. We only prove the first item. For $i \in (m)$, let a_i and b_i be as in Definition 3.8 and similarly let $c_i = \sum_{j \in (i)} z_j$. Let $i \in (m)$ be such that $(a_i - c_i) > 0$, we will show this coordinate contributes at least as much to $\overrightarrow{\mathsf{W1}}(x,y) + \overrightarrow{\mathsf{W1}}(y,z)$ as it does to $\overrightarrow{\mathsf{W1}}(x,z)$ (this concludes the proof, since only positive $(a_i - c_i)$ contributes to $\overrightarrow{\mathsf{W1}}(x,z)$). Assuming that both $(a_i - b_i)$ and $(b_i - c_i)$ are positive, then this coordinates contributes $\frac{1}{m} \cdot ((a_i - b_i) + (b_i - c_i)) = \frac{1}{m} \cdot (a_i - c_i)$ to $\overrightarrow{\mathsf{W1}}(x,y) + \overrightarrow{\mathsf{W1}}(y,z)$. In the case that one of $(a_i - b_i)$ and $(b_i - c_i)$ is positive and the other is negative, then the only (positive) term that contributes to $\overrightarrow{\mathsf{W1}}(x,y) + \overrightarrow{\mathsf{W1}}(y,z)$ is larger than $\frac{1}{m} \cdot (a_i - c_i)$.

□

3.4.3 Shift distance

Suppose we have a set of empirical examples $x_1, \ldots, x_\ell \leftarrow \mathcal{S}$. Let $u(i) = \mathsf{Bin}(x_i)$ and $v(i)$ be the "claimed" value for $u(i)$ (which might differ from the honest bin labels $\mathsf{Bin}(x_i) = u(i)$). Let h^v be the histogram induced by the (possibly false) bin labels $v(i)$, and let h^u be the histogram induced by the true bin labels $\mathsf{Bin}(x_i) = u(i)$. $\mathsf{W1}(h^u, h^v)$ gives the minimal amount of work to move the histogram h^u to the histogram h^v, but notice that the labeling of $x_1, \ldots, x_\ell$ in fact implies a *specific* way of moving from h^u into h^v; namely, for each i such that $v(i) \neq u(i) = \mathsf{Bin}(x_i)$, we need to move one unit of mass from the $u(i)$'th bin to the $v(i)$'th bin. The following shift distance captures this notion of the amount of work required by a *specific* way to move from one histogram to another.

Definition 3.10 (Shift distance)**.** *Given two mappings u, v from $[\ell]$ to (m), we define the* right shift distance *as* $\overrightarrow{\mathsf{SH}}(u,v) = \frac{1}{m\ell} \cdot \sum_{i:\, u(i) < v(i)} (v(i) - u(i))$*, the* left shift distance *as* $\overleftarrow{\mathsf{SH}}(u,v) = \frac{1}{m\ell} \cdot \sum_{i:\, u(i) > v(i)} (u(i) - v(i))$*, and the* shift distance *as* $\mathsf{SH}(u,v) = \overrightarrow{\mathsf{SH}}(u,v) + \overleftarrow{\mathsf{SH}}(u,v)$.

We will use the following simple proposition.

Proposition 3.11. *Let u and v be two mappings from $[\ell]$ to (m), then the following holds.*

1. *If $u(i) \leq v(i) + k$ for all $i \in [\ell]$, then $\overleftarrow{\mathsf{SH}}(u,v) \leq k/m$. Similarly if $v(i) - k \leq u(i)$ for all $i \in [\ell]$, then $\overrightarrow{\mathsf{SH}}(u,v) \leq k/m$.*
2. *If $\mathsf{SH}(u,v) \leq k/m$, then for at least $(1-\delta)$ fraction of $i \in [\ell]$ it holds that $|u(i) - v(i)| \leq k/\delta$.*
3. *It holds that $\overrightarrow{\mathsf{SH}}(u,v) = \frac{1}{m\ell} \cdot \sum_{j \in (m)} (|\{i \colon u(i) \leq j \wedge v(i) > j\}|)$, and similarly $\overleftarrow{\mathsf{SH}}(u,v) = \frac{1}{m\ell} \cdot \sum_{j \in (m)} (|\{i \colon v(i) \leq j \wedge u(i) > j\}|)$.*

Proof Sketch. The first part readily follows from Definition 3.10. For the second part note that $\mathrm{E}_{i \leftarrow [\ell]}[|u(i) - v(i)|] = m \cdot \mathsf{SH}(u,v) \leq k$. So by the Markov inequality it holds that $\Pr_{i \leftarrow [\ell]}[|u(i) - v(i)| > k/\delta] \leq \delta$.

The third part holds because if $v(i) > u(i)$, then the index i contributes $\frac{1}{m\ell} \cdot (v(i) - u(i))$ to $\overrightarrow{\mathsf{SH}}(u,v)$ while it contributes $\frac{1}{m\ell}$ to the right hand side for each j such that $u(i) \leq j < v(i)$.

□

3.4.4 Shift distance to Wasserstein distance

The following lemma relates the Wasserstein distance of two histograms to the Shift distance of the samples that these histograms correspond to. It is proven in Appendix A.

Lemma 3.12 (Bounding $\mathsf{W1}$ with SH)**.** *Let u and v be two mappings from $[\ell]$ to (m) and let $h^u = \mathsf{Hist}(u)$ and $h^v = \mathsf{Hist}(v)$. The following holds.*

1. $\overrightarrow{\mathsf{W1}}(h^u, h^v) \leq \overrightarrow{\mathsf{SH}}(u, v)$ *and* $\overleftarrow{\mathsf{W1}}(h^u, h^v) \leq \overleftarrow{\mathsf{SH}}(u, v)$ *(and thus* $\mathsf{W1}(h^u, h^v) \leq \mathsf{SH}(u, v)$*).*
2. $\overrightarrow{\mathsf{W1}}(h^u, h^v) \geq \overrightarrow{\mathsf{SH}}(u, v) - \overleftarrow{\mathsf{SH}}(u, v)$ *and* $\overleftarrow{\mathsf{W1}}(h^u, h^v) \geq \overleftarrow{\mathsf{SH}}(u, v) - \overrightarrow{\mathsf{SH}}(u, v)$*, (and thus* $\mathsf{W1}(h^u, h^v) \geq |\overrightarrow{\mathsf{SH}}(u, v) - \overleftarrow{\mathsf{SH}}(u, v)|$*).*

3.5 AM Languages

A language L is in $\mathbf{AM}[k]$ if there exists a k-round interactive protocol for deciding L where the verifier V is efficient (i.e., polynomially bounded) and public coin (its messages are simply random coins). Namely, for every $x \in L \cap \{0,1\}^n$ it holds that $\Pr[\mathsf{V} \text{ accepts in } \langle \mathsf{P}, \mathsf{V} \rangle(x)] \geq 1 - 2^{-n}$ and for every $x \in \{0,1\}^n \setminus L$ and any cheating (possibly inefficient) prover P^* it holds that $\Pr[\mathsf{V} \text{ accepts in } \langle \mathsf{P}^*, \mathsf{V} \rangle(x)] \leq 2^{-n}$. If $k = 2$, then we simply write $L \in \mathbf{AM}$ (and call L an $\mathbf{AM}$ set or language). Finally, we abuse notation and say that (P, V) is an $\mathbf{AM}[k]$ protocol if it is a k-round, public-coin protocol with an efficient verifier.

We also consider the "promise problem" variant of an $\mathbf{AM}$ language.

Definition 3.13. *Let $\mathsf{M} = (\mathsf{P}, \mathsf{V})$ be an $\mathbf{AM}$ protocol. We say that M is a proof system for the* promise problem *$(\mathcal{Y}, \mathcal{N})$ (where $\mathcal{Y} \cap \mathcal{N} = \varnothing$) if the following holds.*

- $\mathcal{Y} = \bigcup_n \mathcal{Y}_n$ *where* $\mathcal{Y}_n = \{x \in \{0,1\}^n \colon \Pr[\mathsf{V} \text{ accepts in } \langle \mathsf{P}, \mathsf{V} \rangle(x)] \geq 1 - 2^{-n}\}$.
- $\mathcal{N} = \bigcup_n \mathcal{N}_n$ *where* $\mathcal{N}_n = \{x \in \{0,1\}^n \colon \forall \mathsf{P}^*, \Pr[\mathsf{V} \text{ rejects in } \langle \mathsf{P}^*, \mathsf{V} \rangle(x)] \geq 1 - 2^{-n}\}$.

We call $\mathcal{Y}_n$ the set of YES instances, $\mathcal{N}_n$ the set of NO instances and $\mathcal{T}_n = \{0,1\}^n \setminus (\mathcal{Y}_n \cup \mathcal{N}_n)$ the set of non-promise *instances of length n. We also let $\mathcal{T} = \bigcup_n \mathcal{T}_n$ to be the set of all non-promise inputs to M.*

Note that a language $L \in \mathbf{AM}$ iff there exist a two-round public-coin protocol (with an efficient verifier) $\mathsf{M} = (\mathsf{P}, \mathsf{V})$ with an empty non-promise set $\mathcal{T} = \varnothing$.

3.5.1 Error reduction AM protocols

The definition above assumes that the probability a YES instance is accepted is $\geq 1 - 2^{-n}$ and the probability that a NO instance is accepted is $\leq 2^{-n}$. This is equivalent to YES instances being accepted with probability $\geq 2/3$ and NO instances being accepted with probability $\leq 1/3$ for by the following well-known lemma.

Lemma 3.14 (Error reduction for $\mathbf{AM}$ protocols, [6])**.** *Suppose with an efficient verifier and let $\alpha = \alpha(n) < 1, \beta = \beta(n) < 1$ be such that $\alpha - \beta > 1/\operatorname{poly}(n)$. Suppose we define the sets $\mathcal{Y}^\alpha$ and $\mathcal{N}^\beta$ as follows.*

- $\mathcal{Y}^\alpha_n = \{x \in \{0,1\}^n \colon \Pr[\mathsf{V} \text{ accepts in } \langle \mathsf{P}, \mathsf{V} \rangle(x)] \geq \alpha\}$.

- $\mathcal{N}_n^\beta = \{x \in \{0,1\}^n : \forall \mathsf{P}^*,\ \Pr[\mathsf{V} \textit{ accepts in } \langle \mathsf{P}^*, \mathsf{V}\rangle(x)] \leq \beta\}$.

Then there exists an **AM** *protocol* $\mathsf{M}' = (\mathsf{P}', \mathsf{V}')$ *(with an efficient verifier) such that for all* n *the following holds*

- $\forall x \in \mathcal{Y}_n^\alpha$, $\Pr[\mathsf{V}' \textit{ accepts in } \langle \mathsf{P}', \mathsf{V}'\rangle(x)] \geq 1 - 2^{-2n}$.
- $\forall x \in \mathcal{N}_n^\beta, \forall \mathsf{P}^*,\ \Pr[\mathsf{V}' \textit{ rejects in } \langle \mathsf{P}^*, \mathsf{V}'\rangle(x)] \geq 1 - 2^{-2n}$.

3.5.2 Efficient provers for AM protocols

The following lemma says that the prover in any **AM** protocol can be replaced by a $\mathbf{BPP}^{\mathbf{NP}}$ prover, with a loss in the completeness error that depends on the round complexity. It is prover in Appendix A.

Lemma 3.15 ($\mathbf{BPP}^{\mathbf{NP}}$ provers for **AM** protocols)**.** *Let* $\mathsf{M} = (\mathsf{P}, \mathsf{V})$ *be an* $\mathbf{AM}[2k]$ *protocol (with no input) for* $k = O(1)$ *such that*

$$\Pr[\mathsf{V} \textit{ accepts in } \langle \mathsf{P}, \mathsf{V}\rangle] \geq 1 - \delta,$$

for $\delta \geq 1/\operatorname{poly}(n)$*. Then there is a* $\mathbf{BPP}^{\mathbf{NP}}$ *prover strategy* P' *such that*

$$\Pr[\mathsf{V} \textit{ accepts in } \langle \mathsf{P}', \mathsf{V}\rangle] \geq 1 - 2k\delta^{1/2^k}.$$

3.6 Set Size-Estimation and Sampling Protocols

We call a family of sets $\{\mathcal{S}_n\}$ (non-uniformly) efficiently decidable, if there exist a family of polynomial size circuits Boolean $\{C_n\}$ such that $\mathcal{S}_n = \{x \mid C_n(x) = 1\}$. When it is clear from the context we simply write $\mathcal{S}$ instead of $\mathcal{S}_n$.

The following fundamental lemma by [30] provides a protocol to lower-bound the size of efficiently decidable sets up to small multiplicative factor.

Lemma 3.16 (Set lower-bound protocol, [30] Section 4.1)**.** *There exists an* **AM** *protocol* $\mathsf{SetLB} = (\mathsf{P}_{\mathsf{LB}}, \mathsf{V}_{\mathsf{LB}})$*, where the parties get as input an efficiently decidable set* $\mathcal{S} \subseteq \{0,1\}^n$*,*[6] s *(as size of* $\mathcal{S}$*),* ε *(as the approximation parameter), the verifier runs in time* $\operatorname{poly}(n, 1/\varepsilon)$ *and the following holds.*

Completeness. *If* $|\mathcal{S}| \geq s$*, then* $\Pr[\mathsf{V}_{\mathsf{LB}} \textit{ accepts in } \langle \mathsf{P}_{\mathsf{LB}}, \mathsf{V}_{\mathsf{LB}}\rangle(\mathcal{S}, s, \varepsilon)] \geq 1 - 2^{-n}$.

Soundness. *If* $|\mathcal{S}| \leq (1-\varepsilon) \cdot s$*, then for every prover* P^**, it holds that*
$\Pr[\mathsf{V}_{\mathsf{LB}} \textit{ accepts in } \langle \mathsf{P}^*, \mathsf{V}_{\mathsf{LB}}\rangle(\mathcal{S}, s, \varepsilon)] < 2^{-n}$.

We will need a variation of the protocol of Lemma 3.16 over the promise languages. In fact the exact same protocol given in [30] (with the help of the amplification of Lemma 3.14) proves *both* Lemma 3.16 and the following Lemma 3.17. The protocol of [30] is described for an **AM** set $\mathcal{S}$ which clearly contain the efficiently decidable $\mathcal{S}$'s (i.e. Lemma 3.16) as a special case. However the same protocol (and in fact even the same analysis) given in [30], when considered over **AM** *promise* languages yields the following.

[6] $\mathcal{S}$ can be represented as its deciding circuit, or in case of more succinct representations $(\mathsf{P}_{\mathsf{LB}}, \mathsf{V}_{\mathsf{LB}})$ can directly depend on the deciding algorithm of $\mathcal{S}$ when it is uniform.

Lemma 3.17 (Generalized set lower-bound protocol)**.** [30, Section 4.1] *Let* $\mathsf{M} = (\mathsf{P}, \mathsf{V})$ *be an* **AM** *protocol with YES instances* $\{\mathcal{Y}_n\}$ *and non-promise instances* $\{\mathcal{T}_n\}$*. Then there exists an* **AM** *protocol* $\mathsf{GeneralizedSetLB} = (\mathsf{P}_{\mathsf{GLB}}, \mathsf{V}_{\mathsf{GLB}})$*, where the parties get as input* s *(as size of* $\mathcal{Y}_n$*),* ε *(as the approximation parameter), the verifier runs in time* $\mathrm{poly}(n/\varepsilon)$ *and the following holds.*

Completeness. *If* $|\mathcal{Y}_n| \geq s$*, then* $\Pr[\mathsf{V}_{\mathsf{GLB}}$ *accepts in* $\langle \mathsf{P}_{\mathsf{GLB}}, \mathsf{V}_{\mathsf{GLB}} \rangle (\mathsf{M}, s, \varepsilon)] \geq 1 - 2^{-n}$*.*

Soundness. *If* $|\mathcal{Y}_n \cup \mathcal{T}_n| \leq (1-\varepsilon) \cdot s$*, then for every prover* P^**, it holds that* $\Pr[\mathsf{V}_{\mathsf{GLB}}$ *accepts in* $\langle \mathsf{P}^*, \mathsf{V}_{\mathsf{GLB}} \rangle (\mathsf{M}, s, \varepsilon)] < 2^{-n}$*.* [7]

The following lemma and its proof are an adaptation of those of [28, Lemma A.2] for the setting where we are only given an approximation of the size of the set to be sampled from. ([3] also describes a candidate protocol without the proof.) We give a proof in Appendix A.

Lemma 3.18 (Uniform sampling protocol, [28, 3])**.** *There exists an* **AM** *protocol* $\mathsf{UniformSamp}$*, between an* $\mathbf{P}^{\mathbf{NP}}$ *prover* P_{US} *and an efficient verifier* V_{US}*, whose parties get as input, an efficiently decidable set* $\mathcal{S} \subseteq \{0,1\}^n$*,* $s \in \mathbb{N}$*, and an error parameter* $\delta < 1$*, such that the following holds: assuming that* $s \in [(1 \pm \delta/32) \cdot |\mathcal{S}|]$*, then the verifier runs in* $\mathrm{poly}(n, 1/\delta)$ *and either rejects by outputting* $x = \perp$ *or outputs an element* $x \in \mathcal{S}$ *such that:*

Completeness. V_{US} *rejects in* $\langle \mathsf{P}_{\mathsf{US}}, \mathsf{V}_{\mathsf{US}} \rangle (\mathcal{S}, s, \delta)$ *with probability at most* δ*.*

Soundness. $\Delta(x, U_{\mathcal{S}}) \leq \Pr[\mathsf{V}_{\mathsf{US}}$ *rejects in* $\langle \mathsf{P}^*, \mathsf{V}_{\mathsf{US}} \rangle (\mathcal{S}, s, \delta)] + \delta$*, for any (unbounded) prover* P^**.*

4 Sampling with Size and Verify Histogram Protocols

In this section we formally state and prove Lemma 1.4 ($\mathsf{SampleWithSize}$), its extension to the multi-query case (see discussion below) and Lemma 2.1 ($\mathsf{VerifyHist}$). We first prove a "weak" version of Lemma 2.1 (see Lemma 4.5), use this weak version for proving Lemma 1.4 and its multi-query case (formally stated as Lemma 4.1 and Lemma 4.2), and then prove Lemma 2.1 (stated as Lemma 4.4) using Lemma 4.2. In order to keep the text simple, we state the lemmas with respect to **PSPACE** provers, and only give a variant of Lemma 4.2 (Corollary 4.3) with a $\mathbf{BPP}^{\mathbf{NP}}$ prover. The very same approach, however, can be applied to give similar variant of all other lemmas in this section. Let us start with the formal statements of the main lemmas and of Corollary 4.3.

Lemma 4.1 (Restating Lemma 1.4)**.** *There exists an* $\mathbf{AM}[O(1)]$ *protocol* $\mathsf{SampleWithSize} = (\mathsf{P}_{\mathsf{SWS}}, \mathsf{V}_{\mathsf{SWS}})$*, whose parties get as input an efficiently decidable set* $\mathcal{S} \subseteq \{0,1\}^n$*, an accuracy parameter* $\delta \geq 1/\mathrm{poly}(n)$*,* $s \in \mathbb{N}$ *(as size of* $\mathcal{S}$*) and an efficiently computable function* $f : \mathcal{S} \to \{0,1\}^*$*. At the end of the protocol* $\mathsf{V}_{\mathsf{SWS}}$ *either rejects (signified by outputting a special symbol* $x = \perp$*), or it outputs a pair* (x, s_x)*. Assuming that* $s \in [(1 \pm \gamma) \cdot |\mathcal{S}|]$ *for* $\gamma := (\frac{\delta}{10n})^8$*, then the following conditions hold:*

Completeness: $\mathsf{V}_{\mathsf{SWS}}$ *accepts when interacting with* $\mathsf{P}_{\mathsf{SWS}}$ *with probability at least* $1 - \delta$*, and if not rejecting it holds that* $s_x = |f^{-1}(f(x))|$*.*

Soundness: *For all provers* P^**, the following hold:*

[7] Note that M is not really an extra input and the protocol $\mathsf{GeneralizedSetLB}$ implicitly depends on M, but we still indicate it as so for the sake of clarity.

- $\Pr[\mathsf{V}_{\mathsf{SWS}}$ *does not reject in* $\langle \mathsf{P}^*, \mathsf{V}_{\mathsf{SWS}} \rangle \wedge s_x \notin [(1 \pm \delta) \cdot |f^{-1}(f(x))|]] \leq \delta$,
- $\Delta(x, U_{\mathcal{S}}) \leq \delta + \Pr[\mathsf{V}_{\mathsf{SWS}}$ *rejects in* $\langle \mathsf{P}^*, \mathsf{V}_{\mathsf{SWS}} \rangle]$.

For proving Theorem 1.1 we need to handle multiple queries to protocol SampleWithSize simultaneously, using a single constant round protocol, while enforcing independence between the verifiers' outputs in the different calls. While applying Lemma 4.1 independently in parallel for each of the queries does not guarantee such independence (nothing prevents outputs of the different calls to be dependant), a bit more careful usage of this lemma does yield the desired guarantee, which is formally state as the next lemma.

Lemma 4.2 (Multi-query variant of Lemma 4.1)**.** *There exists an* $\mathbf{AM}[O(1)]$ *protocol* $\mathsf{MultSampleWithSize} = (\mathsf{P}_{\mathsf{MSWS}}, \mathsf{V}_{\mathsf{MSWS}})$*, whose parties get as input a tuple of* k *triplets* $((\mathcal{S}_1, s_1, f_1), \ldots, (\mathcal{S}_k, s_k, f_k))$*, where each* $\mathcal{S}_i \subseteq \{0,1\}^{n_i}$ *is efficiently decidable, an accuracy parameter* $\delta \geq 1/\operatorname{poly}(n)$ *for* $n = \sum_i n_i$*,* $s_i \in \mathbb{N}$ *and each* $f_i : \mathcal{S}_i \to \{0,1\}^*$ *is an efficiently computable function. At end of the protocol* $\mathsf{V}_{\mathsf{MSWS}}$ *either rejects (signified by outputting a special symbol* $\perp$*) or outputs* $((x_1, s_{x_1}), \ldots, (x_k,))$*. Assuming that* $s_i \in [(1 \pm \gamma) \cdot |\mathcal{S}_i|]$ *for every* $i \in [k]$*, where* $\gamma := \frac{1}{8k} \cdot (\frac{\delta}{20n})^8$*, then the following holds:*

Completeness: $\mathsf{V}_{\mathsf{MSWS}}$ *accepts when interacting with* $\mathsf{P}_{\mathsf{MSWS}}$ *with probability at least* $1 - \delta$*, and if not rejecting it holds that* $s_{x_i} = \left|f_i^{-1}(f_i(x_i))\right|$ *for every* $i \in [k]$*.*

Soundness: *The following holds for any (unbounded) prover* P^**:*

- $\Pr\left[\mathsf{V}_{\mathsf{MSWS}}\right.$ *does not reject in* $\langle \mathsf{P}^*, \mathsf{V}_{\mathsf{MSWS}} \rangle \wedge \exists i \in [k] \colon s_{x_i} \notin [(1 \pm \delta) \cdot \left|f_i^{-1}(f_i(x_i))\right|]] < \delta$,
- $\Delta((U_{\mathcal{S}_1}, \ldots, U_{\mathcal{S}_k}), (x_1, \ldots, x_k)) \leq \delta + \Pr[\mathsf{V}_{\mathsf{MSWS}}$ *rejects in* $\langle \mathsf{P}^*, \mathsf{V}_{\mathsf{MSWS}} \rangle]$.

Corollary 4.3 (Lemma 4.2 with $\mathbf{BPP}^{\mathbf{NP}}$ prover)**.** *Let* $((\mathcal{S}_1, s_1, f_1), \ldots, (\mathcal{S}_k, s_k, f_k))$ *and* δ *be as in Lemma 4.2. There exists an* $\mathbf{AM}[O(1)]$ *protocol* $\mathsf{MultSampleWithSize} = (\mathsf{P}_{\mathsf{MSWS}}, \mathsf{V}_{\mathsf{MSWS}})$*, where* $\mathsf{P}_{\mathsf{MSWS}}$ *in* $\mathbf{BPP}^{\mathbf{NP}}$*, whose parties get as input* $((\mathcal{S}_1, s_1, f_1), \ldots, (\mathcal{S}_k, s_k, f_k))$ *and* δ*, at end of the protocol* $\mathsf{V}_{\mathsf{MSWS}}$ *either rejects or outputs* $((x_1, s'_1), \ldots, (x_k, s'_k))$*, and the following holds for a universal constant* $c > 1$*. Assuming that* $s_i \in [(1 \pm \gamma) \cdot |\mathcal{S}_i|]$ *for every* $i \in [k]$*, where* $\gamma := \frac{\delta^c}{ckn^8}$*, then*

Completeness: $\mathsf{V}_{\mathsf{MSWS}}$ *accepts when interacting with* $\mathsf{P}_{\mathsf{MSWS}}$ *with probability at least* $1 - \delta$*.*

Soundness: *The following holds for any (unbounded) prover* P^**:*

- $\Pr\left[\mathsf{V}_{\mathsf{MSWS}}\right.$ *does not reject in* $\langle \mathsf{P}^*, \mathsf{V}_{\mathsf{MSWS}} \rangle \wedge \exists i \in [k] \colon s_{x_i} \notin [(1 \pm \delta) \cdot \left|f_i^{-1}(f_i(x_i))\right|]] < \delta$,
- $\Delta((U_{\mathcal{S}_1}, \ldots, U_{\mathcal{S}_k}), (x_1, \ldots, x_k)) \leq \delta + \Pr[\mathsf{V}_{\mathsf{MSWS}}$ *rejects in* $\langle \mathsf{P}^*, \mathsf{V}_{\mathsf{MSWS}} \rangle]$.

Proof. Let c' be the number of rounds of Protocol MultSampleWithSize of Lemma 4.1, and let δ' be such that $\delta = 2c'\delta'^{1/2^{c'}}$. Lemma 4.1 yields that if $\gamma \leq \frac{1}{8k} \cdot (\frac{\delta'}{20n})^8$ then both the completeness and the soundness conditions hold with a **PSPACE** prover and the accuracy parameter δ'. Lemma 3.15 yields that we can get a $\mathbf{BPP}^{\mathbf{NP}}$ prover that makes the verifier accept with probability $1 - 2c'\delta'^{1/2^{c'}} = 1 - \delta$. Therefore the honest prover can be implemented in $\mathbf{BPP}^{\mathbf{NP}}$ if γ is small enough:

$$\gamma \leq \frac{1}{8k} \cdot (\frac{\delta'}{200n})^8 = \frac{1}{8k} \cdot \left(\frac{(\delta/2c')^{2^{c'}}}{20n}\right)^8$$

which is implied by $\gamma \leq \frac{\delta^c}{ckn^8}$ for a large enough constant c. □

Lemma 4.4 (Restating Lemma 2.1)**.** *There exists an* $\mathbf{AM}[O(1)]$ *protocol* $\mathsf{VerifyHist} = (\mathsf{P_{VH}}, \mathsf{V_{VH}})$*, whose parties get as input an efficiently decidable set* $\mathcal{S} \subseteq \{0,1\}^n$*,* $s \in \mathbb{N}$ *(as size of* $\mathcal{S}$*), an efficiently computable function* $f : \mathcal{S} \to \{0,1\}^*$*, the histogram parameter* $0 < \varepsilon < 1$ *and a (claimed)* ε*-histogram* $h \in [0,1]^{m+1}$ *for* $m = n/\varepsilon$ *such that the following holds. Let* $h^f \in [0,1]^{m+1}$ *be the (real)* ε*-histogram of* f *with respect to* $\mathcal{S}$*. If* $s \in [(1 \pm \gamma) \cdot |\mathcal{S}|]$ *for* $\gamma = (\frac{\varepsilon}{10n})^{40}$ *then:*

Completeness: *If* $h = h^f$*, then* $\mathsf{V_{VH}}$ *accepts when interacting with* $\mathsf{P_{VH}}$ *with probability at least* $1 - 2^{-n}$*.*

Soundness: *If* $\mathsf{W1}(h^f, h) > 20/m = 20\varepsilon/n$*, then (for any unbounded prover)* $\mathsf{V_{VH}}$ *rejects with probability at least* $1 - 2^{-n}$*.*

We are also using the following "supporting lemmas" whose proofs are given in Section 4.1.

Supporting lemmas

The following protocol is similar to protocol VerifyHist of Lemma 4.4, but provides a weaker guarantee: the prover cannot convince the verifier that $\overrightarrow{\mathsf{W1}}$ is much larger than $\overleftarrow{\mathsf{W1}}$.

Lemma 4.5 (Weak Verify Histogram)**.** *There exists an* $\mathbf{AM}[O(1)]$ *protocol* $\mathsf{WeakVerifyHist} = (\mathsf{P_{WVH}}, \mathsf{V_{WVH}})$*, whose parties get as input an efficiently decidable set* $\mathcal{S} \subseteq \{0,1\}^n$*,* $s \in \mathbb{N}$ *(as size of* $\mathcal{S}$*), an efficiently computable function* $f : \mathcal{S} \to \{0,1\}^*$*, the histogram parameter* $\varepsilon = 1/\operatorname{poly}(n)$ *and a (claimed) histogram* $h \in [0,1]^{m+1}$ *for* $m = n/\varepsilon$ *such that the following holds. Let* $h^f \in [0,1]^{m+1}$ *be the true* ε*-histogram of* f *with respect to* $\mathcal{S}$*. Given the promise that* $s \in [(1 \pm \varepsilon)|\mathcal{S}|]$ *the following holds.*

Completeness: *If* $h = h^f$*, then* $\mathsf{V_{WVH}}$ *accepts when interacting with* $\mathsf{P_{WVH}}$ *with probability at least* $1 - 2^{-n/2}$*.*

Soundness: *If* $\overrightarrow{\mathsf{W1}}(h^f, h) > 4 \cdot \overleftarrow{\mathsf{W1}}(h^f, h) + 100\varepsilon$*, then (for any unbounded prover)* $\mathsf{V_{WVH}}$ *rejects with probability at least* $1 - 2^{-n/2}$*.*

The following lemma states that there exists a protocol that verifies that a claimed empirical labeling is close to a true labeling in the shift distance.

Lemma 4.6 (Verify Empirical Labeling)**.** *There exists an* $\mathbf{AM}[O(1)]$ *protocol* $\mathsf{VerifyEmpLabel} = (\mathsf{P_{VEL}}, \mathsf{V_{VEL}})$*, whose parties get as input an efficiently decidable set* $\mathcal{S} \subseteq \{0,1\}^n$ *(where* $n \geq 20$*), an efficiently computable function* $f : \mathcal{S} \to \{0,1\}^*$*, an histogram parameter* $\varepsilon = 1/\operatorname{poly}(n)$ *and a (claimed) histogram* $h \in [0,1]^{m+1}$*. In addition, the protocol takes as input empirical samples* $x_1, \ldots, x_\ell \in \mathcal{S}$ *along with a (claimed) labeling* $u : [\ell] \to (m)$*.*

Let h^f *be the (true) histogram of* f *over the set* $\mathcal{S}$*. Let* u^f *be the (true) labeling of* x_i *(i.e.,* $u^f(i) = \mathsf{Bin}(x_i)$*) and let* $h^{u^f} = \mathsf{Hist}(u^f)$ *(i.e., the induced histogram of* u^f*).*

Assuming that $s \in [(1 \pm \varepsilon)|\mathcal{S}|]$*,* $\overrightarrow{\mathsf{W1}}(h, h^f) < 4 \cdot \overleftarrow{\mathsf{W1}}(h, h^f) + 100\varepsilon$ *and* $\mathsf{W1}(h^{u^f}, h^f) \leq \varepsilon$*, then the following hold:*

Completeness: *If* $u = u^f$ *and* $\mathsf{W1}(h^u, h) \leq \varepsilon$*, then* $\mathsf{V_{VEL}}$ *accepts when interacting with* $\mathsf{P_{VEL}}$ *with probability at least* $1 - 2^{-n/2}$*.*

Soundness: *If* $\mathsf{SH}(u, u^f) > 111\varepsilon$*, then (for any unbounded prover)* $\mathsf{V_{VEL}}$ *rejects with probability at least* $1 - 2^{-n/2}$*.*

Note about the proofs Throughout our proofs we use the fact that $2^{2\varepsilon} > 1+\varepsilon > 2^{\varepsilon/2}$ and $2^{-\varepsilon/2} > 1-\varepsilon > 2^{-2\varepsilon}$ for small $\varepsilon > 0$. Namely, for small ε enough the range $[1-\varepsilon, 1+\varepsilon]$ is the same as $[2^{-2\varepsilon}, 2^{2\varepsilon}]$ up to a factor of 2 over ε. Therefore we will freely replace expressions of the form $s \in [(1 \pm \varepsilon)|\mathcal{S}|]$ with expressions of the form $s \in [2^{\pm\varepsilon} \cdot |\mathcal{S}|]$ (and sometimes vice versa) to make the manipulations simpler, with the understanding that this affects the statement of our theorems by introducing an inconsequential loss of constant factor 2 in parameters.

The formal proofs of Lemma 4.1 (protocol SampleWithSize) and Lemma 4.4 (protocol VerifyHist) are somewhat different than the overview given in Section 2. The idea, however, remains the same: for SampleWithSize, the prover will claim to us a histogram for D whose correctness we check using image and preimage tests, and then if it is correct we sample many elements plus their sizes from D, check that the empirical histogram of the samples is close to what the prover claimed, and if so output one of them. Instead of using VerifyHist to perform the image and preimage tests, we will define SampleWithSize to explicitly use VerifyEmpLabel to do the preimage tests and WeakVerifyHist to do the image tests, therefore bypassing VerifyHist. In fact, we will then use SampleWithSize to prove VerifyHist (there is no circularity because WeakVerifyHist and VerifyEmpLabel do not depend on VerifyHist). It turns out that for the formal proof, this alternative is cleaner to present, and in fact allows us to achieve better parameters, than proving things in the order presented in Section 2.

Now can now move to proving the main lemmas of this section.

Proof of Lemma 4.1. We first a give an implementation of SampleWithSize and its multi-query case with a **PSPACE** prover. Since they are both constant round protocols Lemma 3.15 yields that there exist an $\mathbf{BPP}^{\mathbf{NP}}$ prover for Protocols SampleWithSize and MultSampleWithSize with slightly weaker guarantee which are good enough for our purposes to prove Theorem 1.1.

Protocol SampleWithSize is defined as follows.

Protocol 4.7.
$\mathsf{SampleWithSize} = (\mathsf{P}_{\mathsf{SWS}}, \mathsf{V}_{\mathsf{SWS}})$.

Common input: An efficiently decidable set $\mathcal{S} \subseteq \{0,1\}^n$, accuracy parameter $\delta = 1/\operatorname{poly}(n)$, an efficiently computable function $f : \mathcal{S} \to \{0,1\}^*$ and a size estimation s (for $|\mathcal{S}|$).

Description: In the following let $\varepsilon = \delta^2/(1000n)$ and $\ell = n/\varepsilon^2$.

1. ***Sampling random points from $\mathcal{S}$.*** *The parties interact, in an execution of the Uniform Sampling Protocol of Lemma 3.18, on input $(\mathcal{S}^\ell, s^\ell, \delta/9)$, where $\mathsf{P}_{\mathsf{SWS}}$ and $\mathsf{V}_{\mathsf{SWS}}$ play the role of the prover and the verifier respectively. Let $(x_1, \ldots, x_\ell)$ denote V_{US}'s output in this execution.*

2. ***Compute the histogram and labeling.*** *$\mathsf{P}_{\mathsf{SWS}}$ computes $h \in [0,1]^{m+1}$ the ε-histogram of f with respect to S as well as the labeling $u\colon [\ell] \mapsto (m)$, where $u(i) = \mathsf{Bin}(x_i)$ (see Definition 3.2 for the definition of $\mathsf{Bin}(x_i)$). $\mathsf{P}_{\mathsf{SWS}}$ sends h, u to $\mathsf{V}_{\mathsf{SWS}}$.*

3. ***Verify the claimed histogram (image tests).*** *The parties interact in the WeakVerifyHist protocol (of Lemma 4.5) on input $(\mathcal{S}, s, f, \varepsilon, h)$ where $\mathsf{P}_{\mathsf{SWS}}$ and $\mathsf{V}_{\mathsf{SWS}}$ play the role of $\mathsf{P}_{\mathsf{WVH}}$ and $\mathsf{V}_{\mathsf{WVH}}$ respectively.*

4. ***Verify the samples (preimage tests).*** *$\mathsf{P}_{\mathsf{SWS}}, \mathsf{V}_{\mathsf{SWS}}$ engage in the VerifyEmpLabel protocol (of Lemma 4.6) on input $(\mathcal{S}, s, f, \varepsilon, h, x_1, \ldots, x_\ell, u)$. If VerifyEmpLabel rejects, then $\mathsf{V}_{\mathsf{SWS}}$ rejects as well.*

5. ***Choose output.*** $\mathsf{V}_{\mathsf{SWS}}$ *chooses* $i \leftarrow [\ell]$ *and outputs* $(x_i, s \cdot 2^{-\varepsilon \cdot u(i)})$.

We next prove the completeness and soundness of Protocol 4.7. Recall our notation that u^f is the honest labeling of the examples $x_1, \ldots, x_\ell$, $h^{u^f} = \mathsf{Hist}(u^f)$ denotes the empirical histogram given by the honest labeling u^f, $h^u = \mathsf{Hist}(u)$ denotes the empirical histogram given by the claimed labeling u, and h^f denotes the honest histogram of f over $\mathcal{S}$. The intuition of the proof roughly follows in three steps, described below.

1. $x_1, \ldots, x_\ell$ are sampled almost uniformly in $\mathcal{S}$, so we can apply a Chernoff bound to argue that $\mathsf{W1}(h^{u^f}, h^f)$ is small.
2. The $\mathsf{WeakVerifyHist}$ protocol guarantees that the claimed histogram h satisfies $\overrightarrow{\mathsf{W1}}(h, h^f) < 4\overleftarrow{\mathsf{W1}}(h, h^f) + 100\varepsilon$, therefore the $\mathsf{VerifyEmpLabel}$ protocol guarantees that $\mathsf{SH}(u, u^f)$ is small.
3. We apply the following lemma that states that if $\mathsf{SH}(u, u^f)$ is small, then by picking a random $i \leftarrow [\ell]$ we get an output $(x_i, s2^{-\varepsilon \cdot u(i)})$ which has the requirements of Lemma 4.1.

Lemma 4.8. *Suppose that* $\mathsf{SH}(u, u^f) \le 111\varepsilon$, *then*

$$\Pr_{i \leftarrow [\ell]}\left[s_i \cdot 2^{-\varepsilon \cdot u(i)} \in \left[|f^{-1}(f(x_i))| \cdot 2^{\pm\delta}\right]\right] \ge 1 - \delta/9.$$

Proof. The promise that $\mathsf{SH}(u, u^f) \le 111\varepsilon = 111n/m$ together with Proposition 3.11 (item 2) yield that $\Pr_{i \leftarrow [\ell]}[|u^f(x_i) - u(x_i)| \le 111n/(\delta/9)] \ge 1 - \delta/9$, and we conclude that

$$\begin{aligned}
& s \cdot 2^{-\varepsilon \cdot u(i)} \\
& \in 2^{\pm\varepsilon} \cdot |\mathcal{S}| \cdot 2^{\varepsilon \cdot u^f(x_i)} \cdot 2^{\pm\varepsilon \cdot (999n/\delta)} && \text{since } s \in [2^{\pm\gamma} \cdot |\mathcal{S}|] \\
& \subseteq 2^{\pm\varepsilon} \cdot 2^{\pm\varepsilon} \cdot |f^{-1}(f(x_i))| \cdot 2^{\pm\varepsilon \cdot (999n/\delta)} && \text{by Proposition 3.4} \\
& = |f^{-1}(f(x_i))| \cdot 2^{\pm\varepsilon \cdot (2 + 999n/\delta)} \\
& \subseteq |f^{-1}(f(x_i))| \cdot 2^{\pm\varepsilon \cdot (1000n/\delta)} \\
& = |f^{-1}(f(x_i))| \cdot 2^{\pm\delta}.
\end{aligned}$$

Thus with probability $1 - \delta/9$ we pick a good i satisfying the requirement of Lemma 4.1. □

We now use Lemma 4.8 to formally prove that $\mathsf{SampleWithSize}$ is complete and sound (with an unbounded prover).

Completeness. If $s \in [2^{\pm\gamma} \cdot |\mathcal{S}|]$, then $s^\ell \in [2^{\pm\gamma\ell} \cdot |\mathcal{S}^\ell|]$. Now note that since $\ell\gamma < (\delta/9)/30$, therefore s satisfies the promise of the Uniform Sampling protocol used in Step 1. By the completeness of Lemma 3.18 the verifier rejects in Step 1 with probability at most $\delta/9$.

Moreover, the uniformity condition of Lemma 3.18 implies that for all sets $T \subseteq \mathcal{S}^\ell$, it holds that

$$\begin{aligned}
& \Pr[(x_1, \ldots, x_\ell) \in T] && (1) \\
& = (\Pr[(x_1, \ldots, x_\ell) \in T] + \Pr[\mathsf{V}_{\mathsf{SWS}} \text{ rejects }] - \Pr[U_{\mathcal{S}^\ell} \in T]) + \Pr[U_{\mathcal{S}^\ell} \in T] - \Pr[\mathsf{V}_{\mathsf{SWS}} \text{ rejects}] \\
& = (\Pr[(x_1, \ldots, x_\ell) \in T \cup \{\perp\}] - \Pr[U_{\mathcal{S}^\ell} \in T \cup \{\perp\}]) + \Pr[U_{\mathcal{S}^\ell} \in T] - \Pr[\mathsf{V}_{\mathsf{SWS}} \text{ rejects}] && (2) \\
& \le \Delta((x_1, \ldots, x_\ell), U_{\mathcal{S}^\ell}) + \Pr[U_{\mathcal{S}^\ell} \in T] - \Pr[\mathsf{V}_{\mathsf{SWS}} \text{ rejects}] && (3) \\
& \le \Pr[U_{\mathcal{S}^\ell} \in T] + \delta/9 && (4)
\end{aligned}$$

In the above, the probability is over the $(x_1, \ldots, x_\ell)$ sampled by $\mathsf{V}_{\mathsf{SWS}}$ as in Step 1, which can possibly be $\perp$ if $\mathsf{V}_{\mathsf{SWS}}$ rejects. In Inequality 2 we use the fact that samples from $U_{\mathcal{S}}$ never equal $\perp$. Inequality 3 follows by the definition of statistical distance. Inequality 4 follows by the uniformity condition of Lemma 3.18.

The following claim asserts that the empirical histogram of uniform samples from $U_{\mathcal{S}^\ell}$ are close to the true histogram with high probability.

Claim 4.9. *Let $T \subseteq \mathcal{S}^\ell$ be the set of tuples $(x_1, \ldots, x_\ell)$ such that the true empirical histogram h^{u^f} satisfies $\mathsf{W1}(h^{u^f}, h^f) > \varepsilon$. Then*

$$\Pr[U_{\mathcal{S}^\ell} \in T] < m2^{-n}$$

Proof. It follows that for each $j \in (m-1)$ that

$$\Pr_{x \leftarrow \mathcal{S}}[\mathsf{Bin}(x) \leq j] = \sum_{i \in (j)} h_i^f$$

Let X_i^j be the random variable such that $X_i^j = 1$ if $\mathsf{Bin}(x_i) \leq j$, and $X_i^j = 0$ otherwise. Since $\sum_{i\in(j)} h_i^{u^f} = \frac{1}{\ell} \cdot \sum_{i\in[\ell]} X_i^j$, applying a Chernoff bound yields that

$$\begin{aligned}
&\Pr\left[|(\sum_{i\in(j)} h_i^{u^f}) - (\sum_{i\in(j)} h_i^f)| > \varepsilon\right] \\
&= \Pr\left[|\tfrac{1}{\ell}\sum_{i=1}^{\ell} X_i^j - (\sum_{i\in(j)} h_i^f)| > \varepsilon\right] \\
&\leq 2e^{-\ell\varepsilon^2} = 2e^{-n} < 2^{-n}.
\end{aligned}$$

It follows that

$$\begin{aligned}
&\Pr[U_{\mathcal{S}^\ell} \in T] \\
&= \Pr[\mathsf{W1}(h^{u^f}, h^f) > \varepsilon] = \Pr\left[\sum_{j\in(m)} |(\sum_{i\in(j)} h_i^{u^f}) - (\sum_{i\in(j)} h_i^f)| > m\varepsilon\right] \\
&\leq \left(\sum_{j\in(m-1)} \Pr\left[|(\sum_{i\in(j)} h_i^{u^f}) - (\sum_{i\in(j)} h_i^f)| > \varepsilon\right]\right) \leq m2^{-n}.
\end{aligned}$$

□

Since the prover is honest, $h = h^f$ and therefore it follows by the completeness of $\mathsf{WeakVerifyHist}$ that $\mathsf{V}_{\mathsf{SWS}}$ rejects in Step 3 the with probability at most $2^{-n/2}$.

Since the prover is honest, $h = h^f$ and $u = u^f$. Therefore by Claim 4.9 and Inequality 4, it holds that $\Pr[(x_1, \ldots, x_\ell) \in T] \leq m2^{-n} + \delta/9$. Therefore, suppose $(x_1, \ldots, x_\ell) \notin T$, which implies that $\mathsf{W1}(h^{u^f}, h^f) \leq \varepsilon$. Along with the promise that $s \in [2^{\pm\gamma} \cdot |\mathcal{S}|] \subseteq [2^{\pm\varepsilon} \cdot |\mathcal{S}|]$ holds, and the fact that $h = h^f$, we can apply the completeness of Lemma 4.6 to deduce that the verifier rejects in Step 4 with probability at most $2^{-n/2}$.

So if the prover is honest the verifier does not reject in any of the steps with probability $\leq 2\delta/9 + 2 \cdot 2^{-n/2} < \delta$.

Soundness. Fix a cheating prover P^*. Let $\mathsf{BadSize}$ be the bad event that $s_i 2^{-\varepsilon u(i)} \notin [2^{\pm\delta}|f^{-1}(f(x_i))|]$ and let $\mathsf{NoReject}$ be the event that $\mathsf{V}_{\mathsf{SWS}}$ does not reject. Let $\mathsf{NoReject}_i$ be the event that $\mathsf{V}_{\mathsf{SWS}}$ does not reject in Steps $1,\ldots,i$ (but may possibly reject in later steps), and finally let $\mathsf{BadHist}$ be the event that $(x_1,\ldots,x_\ell) \in T$ as defined in Claim 4.9. Our goal is to bound $\Pr[\mathsf{BadSize} \wedge \mathsf{NoReject}]$. It holds that

$$\begin{aligned}
&\Pr[\mathsf{BadSize} \wedge \mathsf{NoReject}] && (5)\\
&\leq \Pr[\mathsf{BadSize} \wedge \mathsf{NoReject}_1]\\
&= \Pr[\mathsf{BadSize} \wedge \mathsf{NoReject}_1 \wedge \mathsf{BadHist}] + \Pr[\mathsf{BadSize} \wedge \mathsf{NoReject}_1 \wedge \overline{\mathsf{BadHist}}]\\
&\leq \Pr[\mathsf{BadHist}] + \Pr[\mathsf{BadSize} \mid \mathsf{NoReject}_1 \wedge \overline{\mathsf{BadHist}}] && (6)
\end{aligned}$$

As in the analysis of completeness, Claim 4.9 and Inequality 4 yield that $\Pr[\mathsf{BadHist}] \leq m2^{-n} + \delta/9$. In order to bound the second probability in Inequality 6, it suffices to bound the sum of the probabilities of the following events conditioned on $\mathsf{NoReject}_1 \wedge \overline{\mathsf{BadHist}}$:

1. The probability that $\overrightarrow{\mathsf{W1}}(h^f,h) > 4\overleftarrow{\mathsf{W1}}(h^f,h) + 100\varepsilon$ and $\mathsf{WeakVerifyHist}$ accepts.
2. Conditioned on all previous events not occuring, the probability that $\mathsf{SH}(u,u^f) > 111\varepsilon$ and $\mathsf{VerifyEmpLabel}$ accepts.
3. Conditioned on all previous events not occuring, the probability that the output $s_i 2^{-\varepsilon u(i)} \notin [2^{\pm\delta}|f^{-1}(f(x_i))|]$ (i.e., $\mathsf{BadSize}$ occurs).

By conditioning on $\mathsf{NoReject}_1 \wedge \overline{\mathsf{BadHist}}$, all the promises required by Lemma 4.5 are satisfied, therefore by the soundness of Lemma 4.5 the first item occurs with probability $< 2^{-n/2}$. Then, in the second item the promises of Lemma 4.6 are satisfied, so the soundness of Lemma 4.6 implies that this event occurs with probability $< 2^{-n/2}$. Finally, the Correctness Lemma 4.8 implies that the last event occurs with probability $\delta/9$. Therefore the conditional probability in Inequality 6 is bounded by $2^{-n/2+1}+\delta/9$, and the entire probability in Inequality 6 is bounded by $m2^{-n}+2\delta/9+2^{-n/2+1} < \delta$.

For the second part of the soundness, let $\mathsf{Reject} = \overline{\mathsf{NoReject}}$ and $\mathsf{Reject}_i = \overline{\mathsf{NoReject}}_i$. By the uniformity guarantee of Lemma 3.18, it holds for every i, that

$$\Delta(x_i, U_{\mathcal{S}}) \leq \Pr[\mathsf{Reject}_1] + \delta/9$$

and it particular it holds if i is chosen at random. Note that $\mathsf{Reject}_i \cap \mathsf{Reject}_j = \varnothing$ for any $i \neq j$. Therefore if as a mental experiment we choose $i \leftarrow [\ell]$ at the beginning (rather than the last step), the final output x satisfies:

$$\begin{aligned}
\Delta(x, U_{\mathcal{S}}) &= \Delta(x_i, U_{\mathcal{S}}) + \Pr[\mathsf{Reject}_2 \vee \mathsf{Reject}_3 \vee \mathsf{Reject}_4]\\
&\leq \Pr[\mathsf{Reject}_1] + \delta/9 + \Pr[\mathsf{Reject}_2 \vee \mathsf{Reject}_3 \vee \mathsf{Reject}_4]\\
&= \delta/9 + \Pr[\mathsf{Reject}]\\
&= p + \delta/9,
\end{aligned}$$

which concludes the proof. □

Next, we prove the multiple-query version of the above lemma.

Proof. (of Lemma 4.2) Protocol $\mathsf{MultSampleWithSize}$ is defined as follows.

Protocol 4.10.
$\mathsf{MultSampleWithSize} = (\mathsf{P}_{\mathsf{MSWS}}, \mathsf{V}_{\mathsf{MSWS}})$.

Common input: An accuracy parameter δ, and for every $i \in [k]$: an efficiently decidable set $\mathcal{S}_i$, efficiently computable function $f_i \colon \mathcal{S}_i \to \{0,1\}^*$ and a size estimation s_i.

Description: In the following let $\mathcal{S} = \mathcal{S}_1 \times \cdots \times \mathcal{S}_k \subseteq \{0,1\}^n$, let $g(x_1, \ldots, x_k) := (f_1(x_1), \ldots, f_k(x_k))$ and let $s = s_1 \cdots s_k$.

1. ***Sampling points jointly.*** *The parties interact in an execution of the* SampleWithSize *protocol of Lemma 4.1 on input $(\mathcal{S}, s, g, \delta/2)$, where $\mathsf{P}_{\mathsf{SWS}}$ and $\mathsf{V}_{\mathsf{SWS}}$ play the role of the prover and the verifier respectively. Let $(x = (x_1, \ldots, x_k), s_x)$ denote $\mathsf{V}_{\mathsf{SWS}}$'s output.*
2. ***Sending the sibling sizes.*** $\mathsf{P}_{\mathsf{MSWS}}$ *sends $s_{x_i} = |f_i^{-1}(f_i(x_i))|$ for every $i \in [k]$ to $\mathsf{V}_{\mathsf{MSWS}}$, and $\mathsf{V}_{\mathsf{MSWS}}$ rejects if $s_x \neq s_{x_1} \cdots s_{x_k}$.*
3. ***Lower-bound test.*** *For every $i \in [k]$ in parallel, the parties interact in an execution of the lower-bound protocol of Lemma 3.16 on input $(f_i^{-1}(f_i(x_i)), s_{x_i}, \delta/8k)$, where $\mathsf{P}_{\mathsf{SWS}}$ and $\mathsf{V}_{\mathsf{SWS}}$ play the role of the prover and the verifier respectively.*
4. ***Output.*** $\mathsf{V}_{\mathsf{SWS}}$ *outputs $((x_1, s_{x_1}), \ldots, (x_k, s_{x_k}))$.*

Since $s_i \in [(1 \pm \gamma) \cdot |\mathcal{S}_i|] \subseteq [2^{\pm 2\gamma} \cdot |\mathcal{S}_i|]$ for every $i \in [k]$, it follows that $s \in [2^{\pm 2k\gamma} \cdot |\mathcal{S}|] \subseteq [(1 \pm 8k\gamma) \cdot |\mathcal{S}|]$. Therefore, s satisfies the promise of Lemma 4.1 for accuracy parameter $\delta/2$ (recall that $\gamma := \frac{1}{8k} \cdot (\frac{\delta}{20n})^8$).

Completeness. When interacting with the honest prover, Lemma 4.1 yields that $\mathsf{V}_{\mathsf{MSWS}}$ rejects in Step 1 with probability at most $\delta/2$. Clearly, $\mathsf{V}_{\mathsf{MSWS}}$ does not reject in Step 2, and Lemma 3.16 and a union bound yield that $\mathsf{V}_{\mathsf{MSWS}}$ rejects in Step 3 with probability at most $k \cdot \mathrm{neg}(n) = \mathrm{neg}(n)$. Therefore, the probability that $\mathsf{V}_{\mathsf{MSWS}}$ rejects is bounded by $\delta/2 + \mathrm{neg}(n) < \delta$.

Soundness. In the following we assume without loss of generality that $s_x = s_{x_1} \cdots s_{x_k}$ (as otherwise $\mathsf{V}_{\mathsf{SWS}}$ rejects). Lemma 4.1 yields that

$$\Pr[\overline{E_1} \wedge \mathsf{V}_{\mathsf{SWS}} \text{ does not reject at Step 1}] < \delta/2, \tag{7}$$

where E_1 is the event that $s_x \in (1 \pm \delta/2) \cdot |g^{-1}(g(x))|$, and Lemma 3.16 yields that

$$\Pr[\overline{E_2} \wedge \mathsf{V}_{\mathsf{SWS}} \text{ does not reject at Step 3}] \leq \mathrm{neg}(n), \tag{8}$$

where E_2 is the event that $s_{x_i} \leq (1 + \delta/8k) \cdot |f_i^{-1}(f_i(x_i))|$ for every $i \in [k]$.

When E_2 occurs, then for every $i \in [k]$ it holds (using the fact that $(1 + \delta/(8k))^{k-1} \leq 1 + \delta/2$) that

$$\Pi_{j \neq i} s_{x_j} \leq (1 + \delta/2) \cdot \Pi_{j \neq i} |f_j^{-1}(f_j(x_j))|$$

Since $|g^{-1}(g(x))| = \Pi_i |f_i^{-1}(f_i(x_i))|$, therefore the following holds for all $i \in [k]$ when $E_1 \wedge E_2$ occurs:

$$s_{x_i} = \frac{s_x}{\prod_{j \neq i} s_{x_i}} \geq \frac{(1 - \delta/2) \cdot \prod_{j=1}^{k} |f_j^{-1}(f_j(x_j))|}{(1 + \delta/2) \prod_{j \neq i} |f_j^{-1}(f_j(x_j))|} \geq (1 - \delta) |f_i^{-1}(f_i(x_i))| \tag{9}$$

Hence it follows that:

$$\begin{aligned}&\Pr[\exists i\colon s_{x_i}\cdot(1\pm\delta)\cdot|f_i^{-1}(f_i(x_i))|\wedge \mathsf{V_{SWS}} \text{ does not reject}]\\&\leq \Pr[\mathsf{V_{SWS}} \text{ does not reject}\wedge\overline{E_1\wedge E_2}]\\&\leq \delta/2+\mathrm{neg}(n)<\delta.\end{aligned}$$

For the second part of the soundness proof, Lemma 4.1 yields that

$$\Delta((U_{\mathcal{S}_1},\ldots,U_{\mathcal{S}_k}),(x_1,\ldots,x_k))\leq \delta/2+\Pr[\mathsf{V_{MSWS}} \text{ rejects in Step 1}].$$

As in the proof ofLemma 4.1, since the event that $\mathsf{V_{MSWS}}$ rejects in later steps is disjoint from the event that $\mathsf{V_{MSWS}}$ rejects in Step 1, it follows that

$$\Delta((U_{\mathcal{S}_1},\ldots,U_{\mathcal{S}_k}),(x_1,\ldots,x_k))\leq \delta/2+\Pr[\mathsf{V_{MSWS}} \text{ rejects}]$$

□

Proof. (of Lemma 4.4) The protocol VerifyHist described below only achieves completeness and soundness $O(\varepsilon)$. This this can be easily amplified, however, to error below 2^{-n} via parallel repetition.

Protocol 4.11. $\mathsf{VerifyHist}=(\mathsf{P_{VH}},\mathsf{V_{VH}})$.

Common input: An efficiently decidable set $\mathcal{S}\subseteq\{0,1\}^n$, an efficiently computable function $f:\mathcal{S}\to\{0,1\}^*$, a size estimation s (for $|\mathcal{S}|$), the histogram parameter $0<\varepsilon<1$ and a (claimed) ε-histogram h of f with respect to $\mathcal{S}$.

Description: Note that it holds that $m=n/\varepsilon$ where $h\in[0,1]^{m+1}$. In the following let $\ell=100m^2n=100n^3/\varepsilon^2$.

1. ***Sample ℓ random elements from $\mathcal{S}$.*** *The parties interact in an execution of the parallel* SampleWithSize *protocol of Lemma 4.2, on input $(\varepsilon,(\mathcal{S}_1,s,f),\ldots,(\mathcal{S}_\ell,s,f))$, where $\mathcal{S}_i=\mathcal{S}$ for all $i\in[\ell]$ and $\mathsf{P_{VH}}$ and $\mathsf{V_{VH}}$ play the role of $\mathsf{P_{MSWS}}$ and $\mathsf{V_{MSWS}}$ respectively. Let $((x_1,s_{x_1}),\ldots,(x_\ell,s_{x_\ell}))$ be the result of the interaction.*
2. ***Approximate the histogram.*** *For $i\in[\ell]$, let $u(i)=\lfloor(\log(s_{x_i}/s))/\varepsilon\rfloor$. Let $h^u=\mathsf{Hist}(u)$ be the empirical histogram concluded from the mapping u according to Definition 3.3.*
3. ***Verify the claim.*** *Reject if $\mathsf{W1}(h,h^u)\geq 10/m$, and accept otherwise.*

Let $h=h^f$, let $u^f(i)=\mathsf{Bin}(x_i)$ indicate the real bin number of x_i according to Definition 3.2, and let h^{u^f} be the empirical histogram concluded from u^f.

Notice that $\gamma\ll\frac{1}{8\ell}\cdot(\frac{\delta}{20n})^8$ and so s is in the right range to be used for MultSampleWithSize protocol in Step 1.

We will show that for any prover strategy with high probability either the verifier rejects or h^u is a "good" approximation for h^f in the 1st Wasserstein distance. Then comparing h^u to h to decide accept or reject (Step 3) works properly.

Let $T_1\subset(\mathcal{S}\times\mathbb{N})^\ell$ be the "bad set" of tuples $((x_1,s_{x_1}),\ldots,(x_\ell,s_{x_\ell}))$ where there exist $i\in[\ell]$ such that $s_{x_i}\notin[2^{\pm\varepsilon}|f^{-1}(f(x_i))|]$.

In the following we assume without loss of generality that the x_i's are elements of $\mathcal{S}$.

Claim 4.12. *If* $((x_1, s_{x_1}), \ldots, (x_\ell, s_{x_\ell})) \notin T_1$ *then* $\mathsf{W1}(h^u, h^{u^f}) \leq 6/m$.

Proof. If $((x_1, s_{x_1}), \ldots, (x_\ell, s_{x_\ell})) \notin T_1$ then for all $i \in [\ell]$ it holds that $s_{x_i} \in [2^{\pm\varepsilon}|f^{-1}(f(x_i))|]$. Therefore

$$\begin{aligned} u(i) &= \lfloor(\log(s_{x_i}/s))/\varepsilon\rfloor \\ &\in [(\log(s_{x_i}/s))/\varepsilon \pm 1] \\ &\subseteq [(\log(2^{\pm\varepsilon}|f^{-1}(f(x_i))|/s))/\varepsilon \pm 1] \\ &= [\pm 1 + (\log(|f^{-1}(f(x_i))|/s))/\varepsilon \pm 1] \\ &\subseteq \{\mathsf{Bin}(x_i) \pm 3\}\,. \end{aligned}$$

But if $u(i) \in \{\mathsf{Bin}(x_i) \pm 3\}$ for all $i \in [\ell]$, then by Proposition 3.11 and Lemma 3.12 it holds that

$$\mathsf{W1}(h^u, h^{u^f}) \leq \mathsf{SH}(u, u^f) = \overleftarrow{\mathsf{SH}}(u, u^f) + \overrightarrow{\mathsf{SH}}(u, u^f) \leq 3/m + 3/m = 6/m. \tag{10}$$

□

Let $T_2 \subseteq \mathcal{S}^\ell$ be another "bad set", the set of tuples $(x_1, \ldots, x_\ell)$ such that $\mathsf{W1}(h^{u^f}, h^f) > 1/m$. The following claim can be proven similar to Claim 4.9:

Claim 4.13. $\Pr[U_{\mathcal{S}^\ell} \in T_2] < m2^{-n} = \mathrm{neg}(n)$.

In the following let $\mathsf{NoReject}_1$ be the event that V_{VH} does not reject in Step 1, let Reject_1 be the event that V_{VH} rejects in Step 1, and let Reject be the event that V_{VH} rejects in some step.

Claim 4.14. *Let* $((x_1, s_{x_1}), \ldots, (x_\ell, s_{x_\ell}))$ *be the result of Step 1. Then it holds that*

$$\Pr[\mathsf{NoReject}_1 \wedge \mathsf{W1}(h^u, h^f) \geq 10/m] \leq 3\varepsilon$$

Proof. Note that if $((x_1, s_{x_1}), \ldots, (x_\ell, s_{x_\ell})) \notin T_1$ and $(x_1, \ldots, x_\ell) \notin T_2$ then by the definition of T_2 and Claim 4.12 it holds that

$$\mathsf{W1}(h^u, h^f) \leq \mathsf{W1}(h^u, h^{u^f}) + \mathsf{W1}(h^{u^f}, h^f) \leq 1/m + 6/m < 10/m.$$

So it holds that:

$$\begin{aligned} &\Pr[\mathsf{NoReject}_1 \wedge \mathsf{W1}(h^u, h^f) \geq 10/m] \\ &\leq \Pr[\mathsf{NoReject}_1 \wedge (((x_1, s_{x_1}), \ldots, (x_\ell, s_{x_\ell})) \in T_1 \vee (x_1, \ldots, x_\ell) \in T_2)] \\ &\leq \Pr[\mathsf{NoReject}_1 \wedge ((x_1, s_{x_1}), \ldots, (x_\ell, s_{x_\ell})) \in T_1] + \Pr[\mathsf{NoReject}_1 \wedge (x_1, \ldots, x_\ell) \in T_2]. \end{aligned}$$

Now Claim 4.14 will follow by proving the following inequalities.

$$\Pr[\mathsf{NoReject}_1 \wedge ((x_1, s_{x_1}), \ldots, (x_\ell, s_{x_\ell})) \in T_1] \leq \varepsilon \tag{11}$$

$$\Pr[\mathsf{NoReject}_1 \wedge (x_1, \ldots, x_\ell) \in T_2] \leq \varepsilon + \mathrm{neg}(n) \tag{12}$$

The soundness of Lemma 4.2 yields that

$$\Pr[\mathsf{V}_{\mathsf{MSWS}} \text{ does not reject} \wedge \exists i \in [\ell] \colon s_{x_i} \notin [2^{\pm\varepsilon}|f^{-1}(f(x_i))|] \leq \varepsilon$$

which is equivalent to Inequality 11.

The soundness of Lemma 4.2 also yields that

$$\Delta(U_{\mathcal{S}^\ell}, (x_1, \ldots, x_\ell)) \leq \varepsilon + \Pr[\mathsf{V}_{\mathsf{MSWS}} \text{ rejects}]. \tag{13}$$

By using Lemma 3.7 over Inequality 13 with parameters $Y = U_{\mathcal{S}^\ell}, X = (x_1, \ldots, x_\ell), T = T_2$ and $\delta = \varepsilon$ we get that

$$\Pr[\mathsf{NoReject}_1 \wedge (x_1, \ldots, x_\ell) \in T_2] \leq \varepsilon + \Pr[U_{\mathcal{S}^\ell} \in T_2].$$

But Claim 4.13 yields that $\Pr[U_{\mathcal{S}^\ell} \in T_2] \leq \text{neg}(n)$ and so Inequality 12 follows. □

Now we will prove that Protocol VerifyHist is complete and sound as described in Lemma 4.4 (we prove this for completeness/soundness error $O(\varepsilon)$, but this can be amplified to 2^{-n} by repeating in parallel).

Completeness. Let the prover be honest and $h = h^f$. By Claim 4.14 it holds that

$$\Pr[\mathsf{Reject}_1 \vee \mathsf{W1}(h^u, h^f) \leq 10/m] \geq 1 - 3\varepsilon.$$

By the completeness of Lemma 4.2, V_{VH} rejects in Step 1 with probability at most $\delta = \varepsilon$ and so

$$\Pr[\mathsf{W1}(h^u, h^f) \leq 10/m] \geq 1 - 4\varepsilon.$$

But if $\mathsf{W1}(h^u, h^f) \leq 10/m$, then V_{VH} does not reject in Step 3. Therefore by a union bound:

$$\Pr[\mathsf{Reject}] \leq \varepsilon + 4\varepsilon = 5\varepsilon.$$

Soundness. Suppose $\mathsf{W1}(h^f, h) > 20/m$.

By Claim 4.14 it holds that

$$\Pr[\mathsf{Reject}_1 \vee \mathsf{W1}(h^u, h^f) \leq 10/m] \geq 1 - 3\varepsilon.$$

But if $\mathsf{W1}(h^f, h^u) < 10/m$, then it would hold that

$$\mathsf{W1}(h^u, h) > \mathsf{W1}(h^f, h) - \mathsf{W1}(h^f, h^u) > 20/m - 10/m = 10/m.$$

So it holds that

$$\Pr[\mathsf{Reject}_1 \vee \mathsf{W1}(h^u, h) > 10/m] \geq 1 - 3\varepsilon.$$

Finally since $\mathsf{W1}(h^u, h) > 10/m$ makes the verifier reject in Step 3 therefore:

$$\Pr[\mathsf{Reject}] \geq 1 - 3\varepsilon.$$

□

4.1 Proving Supporting Lemmas

In this section we prove Lemma 4.5 and Lemma 4.6.

Proof of WeakVerifyHist **Lemma 4.5**

Proof. We will use the following definition:

Definition 4.15 (Exponential sum)**.** *Given $x \in \mathbb{R}^{j+1}$ and $\varepsilon > 0$, we define the* exponential sum *of x as $\mathsf{ES}_\varepsilon(x) = \sum_{i\in(j)} x_i \cdot 2^{i\varepsilon}$ and the* normalized exponential sum *of x as $\widetilde{\mathsf{ES}}_\varepsilon(x) = \mathsf{ES}_\varepsilon(x)/2^{j\varepsilon}$. When ε is clear from the context, we omit ε from the notation.*

If h^f be the ε-histogram of a function f over $\{0,1\}^n$, then by Proposition 3.4, $\mathsf{ES}_\varepsilon(h^f{}_{\leq j})$ lower-bounds $|\bigcup_{i\in(j)} \mathcal{B}_i|$ with approximation factor $2^{\varepsilon j}$.

Protocol 4.16.
$\mathsf{WeakVerifyHist} = (\mathsf{P}_{\mathsf{WVH}}, \mathsf{V}_{\mathsf{WVH}})$.

Common input: An efficiently decidable set $\mathcal{S} \subseteq \{0,1\}^n$, a size estimation s (for $|\mathcal{S}|$), a function $f : \mathcal{S} \to \{0,1\}^*$, a histogram parameter $\varepsilon = 1/\operatorname{poly}(n)$, and h a (claimed) ε-histogram of f over $\mathcal{S}$.

Description: We assume for sake of simplicity assume that $f(\mathcal{S}) \subseteq \{0,1\}^n$. [8] For each $j \in (m)$, let M^j be the following protocol: on input $y \in \{0,1\}^n$, if $y \notin f(\mathcal{S})$, the verifier rejects. Otherwise, the parties engage in the Set-Lower-Bound protocol of Lemma 3.16 with input $(f^{-1}(y), s \cdot 2^{-\varepsilon\cdot(j+2)}, \varepsilon)$. Protocol WeakVerifyHist *is defined as follows:*

1. *The parties interact for every $j \in (m)$, in parallel, in an execution of the of Lemma 3.17, on input $(1^n, \mathsf{M}^j, \mathsf{ES}_\varepsilon(h_{\leq j}), \varepsilon)$, where $\mathsf{V}_{\mathsf{WVH}}$ and $\mathsf{P}_{\mathsf{WVH}}$ play the role of $\mathsf{V}_{\mathsf{GLB}}$ and $\mathsf{P}_{\mathsf{GLB}}$ respectively.*
2. *$\mathsf{V}_{\mathsf{WVH}}$ accepts if all $\mathsf{V}^j_{\mathsf{WVH}}$'s accept.*

We next prove the completeness and soundness of Protocol 4.16. In the following we fix $\mathcal{S}$, f, n, s and ε, let $\mathcal{B}_j$'s be as in Definition 3.2 with respect to to this fixing. Also let $(\mathcal{Y}^j, \mathcal{N}^j, \mathcal{T}^j)$ be, in order, the YES, NO, and non-promise inputs of length n for the protocol M^j according to Definition 3.13 (since n is fixed, we will not write the index n). Finally, let $\mathsf{GeneralizedSetLB}^j$ be the j'th execution of the Generalized Set-Lower-bound protocol done in a random execution of WeakVerifyHist.

Claim 4.17. *It holds that:*

1. $\bigcup_{i\in(j)} \mathcal{B}_i \subseteq \mathcal{Y}^j$, *and*
2. $\mathcal{Y}^j \cup \mathcal{T}^j \subseteq \bigcup_{1\leq i\leq j+3} \mathcal{B}_i$.

Proof. Let $y \in \mathcal{B}_i$. First consider the case that $i \leq j$. Proposition 3.4 yields that $|\mathcal{S}| \cdot 2^{-\varepsilon\cdot(j+1)} \leq |\mathcal{S}| \cdot 2^{-\varepsilon\cdot(i+1)} < |f^{-1}(y)|$, and by the promise on s it holds that $s \cdot 2^{-\varepsilon\cdot(j+2)} \leq |f^{-1}(y)|$. The completeness of Lemma 3.16 yields that verifier accepts in M^j with probability at least $1 - 2^{-n}$ and therefore $\mathcal{B}_i \subseteq \mathcal{Y}^j$.

Now let $i \geq j + 4$. Proposition 3.4 yields that $\left|f^{-1}(y)\right| \leq |\mathcal{S}| \cdot 2^{-\varepsilon i} \leq |\mathcal{S}| \cdot 2^{-\varepsilon\cdot(j+4)}$. Since $|\mathcal{S}| \leq s \cdot 2^\varepsilon$, it holds that $\left|f^{-1}(y)\right| \leq 2^{-\varepsilon}(s \cdot 2^{-\varepsilon\cdot(j+2)})$. Therefore, the soundness of Lemma 3.16

[8] Since f is efficiently computable, it holds that $f(\mathcal{S}) \subseteq \{0,1\}^{\operatorname{poly}(n)}$ and all the proof can easily be adapted to this case as well.

yields that the verifier rejects in M^j with probability at least $1-2^{-n}$, which implies that $\mathcal{B}_i \subseteq \mathcal{N}^j$. Also note that by the definition of M^j, it holds that $\{0,1\}^n \setminus f(\mathcal{S}) \subseteq \mathcal{N}^j$, and therefore

$$\mathcal{Y}^j \cup \mathcal{T}^j \subseteq f(\mathcal{S}) \setminus \bigcup_{j+4\leq i\leq m} \mathcal{B}_i = \bigcup_{1\leq i\leq j+3} \mathcal{B}_i.$$

□

Completeness. Assuming that $h = h^f$, Proposition 3.4 and Claim 4.17 yield that $\mathsf{ES}(h_{\leq j}) = \mathsf{ES}(h^f_{\leq j}) \leq |\bigcup_{i\in(j)} \mathcal{B}_i| \leq |\mathcal{Y}^j|$. Therefore, the completeness of Lemma 3.17 yields that the verifier accepts in $\mathsf{GeneralizedSetLB}^j$ with probability at least $1-2^{-n}$, and by the union bound $\mathsf{V}_{\mathsf{WVH}}$ accepts in all of the $m+1$ instances of $\mathsf{GeneralizedSetLB}^j$'s (simultaneously) with probability at least $1-(m+1)2^{-n} > 1-2^{-n/2}$.

Soundness. The following lemma carries the heart of the proof.

Lemma 4.18. *Let d and d' be two probability distributions over $[0,1]^{m+1}$ and let $\varepsilon, \lambda \in (0,1)$. Assume that*

1. *$m \cdot \varepsilon \geq 1$ and*
2. *$\mathsf{ES}_\varepsilon(d'_{\leq j}) \leq 2^\lambda \cdot \mathsf{ES}_\varepsilon(d_{\leq j})$ for all $j \in (m)$,*

then $\overrightarrow{\mathsf{W1}}(d,d') \leq 16\lambda + 4\overleftarrow{\mathsf{W1}}(d,d')$.

Before proving Lemma 4.18, we first use it to show the soundness of Protocol 4.16. Let $m' = m+3$, $d' = (d'_0,\ldots,d'_{m'}) = (0,0,0,h)$ and $d = (d_0,\ldots,d_{m'}) = (h^f,0,0,0)$ (notice that we are using Lemma 4.18 with dimension $m'+1$ rather than $m+1$). Since $m'\varepsilon > m\varepsilon = n \geq 1$, the first condition of Lemma 4.18 is satisfied. The following claim yields that (d,d') also satisfies the second condition of Lemma 4.18 for the suitable choice of λ.

Claim 4.19. *For any $j \in (m)$ and any prover P^*, either $\mathsf{ES}(h_{\leq j}) \leq 2^{2\varepsilon} \cdot \mathsf{ES}(h^f_{\leq j+3})$ or $\mathsf{V}_{\mathsf{WVH}}$ rejects in $\mathsf{GeneralizedSetLB}^j$ with probability at least $1-2^{-n}$ (where we let $h^f_i = 0$ for $i > m$).*

Proof. Assuming that $\mathsf{ES}(h^f_{\leq j+3}) < 2^{-2\varepsilon} \cdot \mathsf{ES}(h_{\leq j})$, Claim 4.17 together with Proposition 3.4 yield that

$$|\mathcal{Y}^j \cup \mathcal{T}^j| \leq \left|\bigcup_{i\in(j+3)} \mathcal{B}_i\right| < 2^\varepsilon \cdot \mathsf{ES}(h^f_{\leq j+3}) < 2^\varepsilon 2^{-2\varepsilon} \cdot \mathsf{ES}(h_{\leq j}) = 2^{-\varepsilon}\mathsf{ES}(h_{\leq j}).$$

Therefore, Lemma 3.17 yields that $\mathsf{V}_{\mathsf{WVH}}$ rejects in $\mathsf{GeneralizedSetLB}^j$ with probability at least $1-2^{-n}$. □

For $j \in \{0,1,2\}$ and any $\lambda > 0$, it holds that $\mathsf{ES}(d'_{\leq j}) = 0 \leq 2^\lambda \cdot \mathsf{ES}(d_{\leq j})$. Claim 4.19 yields that either $\mathsf{V}_{\mathsf{WVH}}$ rejects with probability at least $1-2^{-n}$ or it holds that

$$\mathsf{ES}(d'_{\leq j}) = 0 + 2^{3\varepsilon} \cdot \mathsf{ES}(h_{\leq j-3}) \leq 2^{3\varepsilon}2^{2\varepsilon}\mathsf{ES}(h^f_{\leq j}) = 2^{5\varepsilon}\mathsf{ES}(d_{\leq j}),$$

for $3 \leq j \leq m'$. Hence, the second requirement of Lemma 4.18 holds for $\lambda = 5\varepsilon$. Below, by $\overrightarrow{\mathsf{W1}}_{m'}(h^f,h)$ we mean the 1st Wasserstein distance when the dimension is increased m' (by adding

three zeros in coordinates $m+1, m+2, m+3$). Note $m' \cdot \mathsf{W1}_{m'}(h^f, h) = m \cdot \mathsf{W1}(h^f, h)$ and that $\mathsf{W1}_{m'}(d', h) \leq 3/m < 3\varepsilon$. We conclude that

$$
\begin{aligned}
&\overrightarrow{\mathsf{W1}}(h^f, h) \\
&= (m'/m)\overrightarrow{\mathsf{W1}}_{m'}(h^f, h) && \text{by change of the dimension } m \to m' \\
&\leq (m'/m)(\overrightarrow{\mathsf{W1}}(h^f, d) + \overrightarrow{\mathsf{W1}}(d, d') + \overrightarrow{\mathsf{W1}}(d', h)) && \text{by Proposition 3.9} \\
&\leq (m'/m)(0 + 16\lambda + 4\overleftarrow{\mathsf{W1}}(d, d') + 3\varepsilon) && \text{by Lemma 4.18} \\
&\leq (m'/m)(16(5\varepsilon) + 4(\overleftarrow{\mathsf{W1}}(d, h^f) + \overleftarrow{\mathsf{W1}}_{m'}(h^f, h) + \overleftarrow{\mathsf{W1}}(h, d')) + 3\varepsilon) && \text{by Proposition 3.9} \\
&= (m'/m) \cdot (83\varepsilon) + 4\overleftarrow{\mathsf{W1}}(h^f, h) && \text{by change of the dimension } m' \to m \\
&\leq 100\varepsilon + 4\overleftarrow{\mathsf{W1}}(h^f, h). && \text{for } m \geq n > 15
\end{aligned}
$$

□

Proof of Lemma 4.18. For the duration of the proof, set $m' = 2m - 1$. We first increase the dimension of d and d' from $m+1$ to $m'+1 = 2m$, by padding them with trailing zeros. Namely, we let $d = (d_0, \ldots, d_m, d_{m+1} = 0, \ldots, d_{m'} = 0)$ and $d' = (d'_0, \ldots, d'_m, d'_{m+1} = 0, \ldots, d'_{m'} = 0)$ (both vectors are now in $[0,1]^{m'+1}$). For $j \in (m')$, let $a_j = \sum_{i \in (j)}(d_i - d'_i)$ and let $a = (a_0, \ldots, a_{m'})$ (note that $a_j = 0$ for $m \leq j \leq m'$). Also we let $a_j = 0$ for $j \notin (m')$ (in particular, $\mathsf{ES}(a_{\leq j}) = 0$ for $j < 0$). The following claim characterizes the difference $\mathsf{ES}(d'_{\leq j}) - \mathsf{ES}(d_{\leq j})$ in terms of the vector a.

Claim 4.20. *For every $j \in (m')$ it holds that $\mathsf{ES}(d'_{\leq j}) - \mathsf{ES}(d_{\leq j}) = (2^\varepsilon - 1) \cdot \mathsf{ES}(a_{\leq j-1}) - 2^{j\varepsilon} a_j$.*

Proof. An intuitive proof is as follows. Consider the process that changes $d_{\leq j}$ into $d'_{\leq j}$ by "pushing" the amount a_i's from d_i to d_{i+1} for every $i \in (j)$. The effect of these changes for $i < j$ on $\mathsf{ES}(d'_{\leq j}) - \mathsf{ES}(d_{\leq j})$ is equal to $-a_i 2^{i\varepsilon} + a_i 2^{(i+1)\varepsilon} = (2^\varepsilon - 1) a_i 2^{i\varepsilon}$ (a_i is removed from d_i and is added to d_{i+1}, where these changes get multiplied by $2^{i\varepsilon}$ and $2^{(i+1)\varepsilon}$ respectively in the exponential sums). For $i = j$ the change to $\mathsf{ES}(d'_{\leq j}) - \mathsf{ES}(d_{\leq j})$ is just the negative part $-a_i 2^{i\varepsilon}$ (a_i is "pushed out" of $(d_1, \ldots, d_i)$). Formally, the proof goes as follows.

$$
\begin{aligned}
&(2^{\varepsilon}-1)\cdot \mathsf{ES}(a_{\le j-1}) - a_j 2^{j\varepsilon} \\
&= \Big(\sum_{i\in(j-1)} (2^{\varepsilon}-1)a_i 2^{i\varepsilon}\Big) - a_j 2^{j\varepsilon} && \text{definition of } \mathsf{ES}(a_{\le j}) \\
&= \Big(\sum_{i\in(j-1)} (2^{\varepsilon}-1)2^{i\varepsilon}\cdot \sum_{k\in(i)} (d_k - d'_k)\Big) - 2^{j\varepsilon}\cdot \sum_{k\in(j)} (d_k - d'_k) && \text{definition of } a_i \\
&= \Big(\sum_{k\in(j-1)} (d_k - d'_k)((2^{\varepsilon}-1)\cdot \sum_{k\le i\le j-1} 2^{i\varepsilon})\Big) - \sum_{k\in(j-1)} (d_k - d'_k)2^{j\varepsilon} - (d_j - d'_j)2^{j\varepsilon} \\
&= \Big(\sum_{k\in(j-1)} (d_k - d'_k)((2^{\varepsilon}-1)\cdot \frac{2^{j\varepsilon}-2^{k\varepsilon}}{(2^{\varepsilon}-1)})\Big) - \sum_{k\in(j-1)} (d_k - d'_k)2^{j\varepsilon} - (d_j - d'_j)2^{j\varepsilon} \\
&= \sum_{k\in(j-1)} (d_k - d'_k)(2^{j\varepsilon} - 2^{k\varepsilon} - 2^{j\varepsilon}) - (d_j - d'_j)2^{j\varepsilon} \\
&= \sum_{k\in(m)} (d_k - d'_k)(-2^{k\varepsilon}) = -(\mathsf{ES}(d_{\le j}) - \mathsf{ES}(d'_{\le j})) = \mathsf{ES}(d'_{\le j}) - \mathsf{ES}(d_{\le j}).
\end{aligned}
$$

□

Note that the second promise of Lemma 4.18 implies that $\mathsf{ES}(d'_{\le j}) - \mathsf{ES}(d_{\le j})$ can not be "too large" because it is bounded by $\le (2^{\lambda}-1)\cdot \mathsf{ES}(d_{\le j})$. The next claim, roughly speaking, asserts that if effect of $(a_1,\ldots,a_{j-1})$ in $\mathsf{ES}(d'_{\le j}) - \mathsf{ES}(d_{\le j})$ (which is captured by $\mathsf{ES}(a_{\le j-1})$) is large enough, then a_j that has a negative effect in $\mathsf{ES}(d'_{\le j}) - \mathsf{ES}(d_{\le j})$ should also be large to compensate the effect of $(a_1,\ldots,a_{j-1})$ for $\mathsf{ES}(d'_{\le j}) - \mathsf{ES}(d_{\le j})$ and keep it 'small". On the other hand, if a_j is large enough then this in turn keeps $\widetilde{\mathsf{ES}}(a_{\le j})$ large. The claim proves the above intuition for the *normalized* exponential sums.

Claim 4.21. *The following holds for every $j \in (m')$:*

1. $a_j \ge (1-2^{-\varepsilon})\cdot \widetilde{\mathsf{ES}}(a_{\le j-1}) - \lambda\cdot \widetilde{\mathsf{ES}}(d_{\le j})$, *and*
2. $\widetilde{\mathsf{ES}}(a_{\le j}) \ge \widetilde{\mathsf{ES}}(a_{\le j-1}) - \lambda\cdot \widetilde{\mathsf{ES}}(d_{\le j})$.

Proof. For every $j \in (m')$, the second promise of Lemma 4.18 implies that $\mathsf{ES}(d'_{\le j}) - \mathsf{ES}(d_{\le j}) \le (2^{\lambda}-1)\cdot \mathsf{ES}(d_{\le j}) < \lambda\cdot \mathsf{ES}(d_{\le j})$. Therefore, Claim 4.20 yields that $(2^{\varepsilon}-1)\cdot \mathsf{ES}(a_{\le j-1}) - 2^{j\varepsilon}a_j < \lambda\cdot \mathsf{ES}(d_{\le j})$, and thus by normalizing the exponential sums we have

$$a_j \ge 2^{-\varepsilon}(2^{\varepsilon}-1)\cdot \widetilde{\mathsf{ES}}(a_{\le j-1}) - \lambda\cdot \widetilde{\mathsf{ES}}(d_{\le j}),$$

which proves the fist part of the claim.

The second part of the claim also holds since

$$
\begin{aligned}
&\widetilde{\mathsf{ES}}(a_{\le j}) \\
&= 2^{-\varepsilon}\cdot \widetilde{\mathsf{ES}}(a_{\le j-1}) + a_j && \text{by definition of } \mathsf{ES}(\cdot) \text{ and } \widetilde{\mathsf{ES}}(\cdot) \\
&\ge 2^{-\varepsilon}\cdot \widetilde{\mathsf{ES}}(a_{\le j-1}) + (1-2^{-\varepsilon})\cdot \widetilde{\mathsf{ES}}(a_{\le j-1}) - \lambda\cdot \widetilde{\mathsf{ES}}(d_{\le j}) && \text{by the first part of Claim 4.21} \\
&= \widetilde{\mathsf{ES}}(a_{\le j-1}) - \lambda\cdot \widetilde{\mathsf{ES}}(d_{\le j}),
\end{aligned}
$$

□

It is clear that $\widetilde{\mathsf{ES}}(d_{\le j}) \le 1$. Trivially it then holds that $\sum_{j\in(m')} \widetilde{\mathsf{ES}}(d_{\le j}) \le m'+1$. The following claim strengthens this trivial bound.

Claim 4.22. *It holds that* $\sum_{j\in(m')} \widetilde{\mathsf{ES}}(d_{\le j}) \le 2/(2^\varepsilon - 1)$.

Proof.

$$\begin{aligned}
&\sum_{j\in(m')} \widetilde{\mathsf{ES}}(d_{\le j}) \\
&= \sum_{j\in(m')} \frac{1}{2^{j\varepsilon}} \cdot \sum_{i\in(j)} d_i 2^{i\varepsilon} = \sum_{i\in(m')} \sum_{i\le j\le m'} d_i 2^{(i-j)\varepsilon} \\
&= \sum_{i\in(m')} d_i \cdot \sum_{k\in(m'-i)} 2^{-k\varepsilon} < \sum_{i\in(m')} d_i \cdot \sum_{k\in(\infty)} 2^{-k\varepsilon} = 2^\varepsilon/(2^\varepsilon-1) < 2/(2^\varepsilon-1).
\end{aligned}$$

□

Recall that Claim 4.21 informally states that if $\widetilde{\mathsf{ES}}(a_{\le j})$ is large for some j, then for $j' > j$, $a_{j'}$ and $\widetilde{\mathsf{ES}}(a_{\le j'})$ are relatively large as well. Looking from the other direction, since eventually we get to the point that $a_{m'} = 0$, none of the $\widetilde{\mathsf{ES}}(a_{\le j})$'s can be too large. This intuition is formalized in the following claim.

Claim 4.23. *For every* $j \in (m')$, *it holds that* $\widetilde{\mathsf{ES}}(a_{\le j}) < 4\lambda/(2^\varepsilon - 1)$.

Proof.

$$\begin{aligned}
&\widetilde{\mathsf{ES}}(a_{\le j}) - 4\lambda/(2^\varepsilon-1) \\
&< \widetilde{\mathsf{ES}}(a_{\le j}) - \lambda \cdot \Big(\sum_{j\le i<m'-1} \widetilde{\mathsf{ES}}(d_{\le i})\Big) - 2\lambda/(2^\varepsilon-1) && \text{Claim 4.22} \\
&\le \widetilde{\mathsf{ES}}(a_{\le m'-1}) - 2\lambda/(2^\varepsilon-1) && \text{induction over the second part of Claim 4.21} \\
&\le \widetilde{\mathsf{ES}}(a_{\le m'-1}) - \frac{\lambda 2^\varepsilon}{(2^\varepsilon-1)} \cdot \widetilde{\mathsf{ES}}(d_{\le m'}) && \text{by } 2^\varepsilon < 2 \text{ and } \widetilde{\mathsf{ES}}(d_{\le m'}) \le 1 \\
&\le \widetilde{\mathsf{ES}}(a_{\le m'-1}) - \frac{\lambda}{(1-2^{-\varepsilon})} \cdot \widetilde{\mathsf{ES}}(d_{\le m'}) \\
&\le a_{m'}/(1-2^{-\varepsilon}) = 0. && \text{the first part of Claim 4.21}
\end{aligned}$$

□

Recall that the conclusion of Lemma 4.18 states that if $\overrightarrow{\mathsf{W1}}(d,d') = \sum_{a_i>0} a_i$ is large, then $\overleftarrow{\mathsf{W1}}(d,d') = -\sum_{a_i<0} a_i$ is large too. In Claim 4.23, we showed that $\widetilde{\mathsf{ES}}(a_{\le j})$'s cannot be too large. Roughly speaking (see the calculation below), large $\sum_{a_i>0} a_i$ makes $\widetilde{\mathsf{ES}}(a_{\le j})$ large, where large $-\sum_{a_i<0} a_i$ makes $\widetilde{\mathsf{ES}}(a_{\le j})$ small. Thus, in order for the claimed bound on $\widetilde{\mathsf{ES}}(a_{\le j})$ to hold, $-\sum_{a_j<0} a_j$ should cancel $\sum_{a_j>0} a_j$. Formally, we first show that:

$$\begin{aligned}
\sum_{j\in(m')} \widetilde{\mathsf{ES}}(a_{\le j}) &= \sum_{j\in(m')} \sum_{i\in(m)} a_i 2^{(i-j)\varepsilon} \\
&= \sum_{i\in(m')} a_i \cdot \sum_{k\in(m'-i)} 2^{-k\varepsilon}
\end{aligned}$$

Therefore from Claim 4.23 and the fact that $m' = 2m - 1$ we conclude that

$$\sum_{i\in(m')} a_i \cdot \sum_{k\in(m'-i)} 2^{-k\varepsilon} \leq 4m'\lambda(2^\varepsilon - 1) < 8m\lambda(2^\varepsilon - 1) \tag{14}$$

On the other hand, since we assumed $m\varepsilon \geq 1$, for every $0 \leq i < m$ we can get the following upper and lower-bounds for $\sum_{k\in(m'-i)} 2^{-k\varepsilon}$ (i.e. the coefficient of a_i in Equation 14):

$$\frac{1}{2(2^\varepsilon - 1)} < \frac{(2^\varepsilon - 2^{-m\varepsilon})}{(2^\varepsilon - 1)} = \sum_{k\in(m)} 2^{-k\varepsilon} \leq \sum_{k\in(m'-i)} 2^{-k\varepsilon} < \sum_{k\in(\infty)} 2^{-k\varepsilon} = \frac{2^\varepsilon}{2^\varepsilon - 1} < \frac{2}{(2^\varepsilon - 1)} \tag{15}$$

By substituting the coefficient of the a_i's in Equation 14 with the proper upper and lower bound of Equation 15 (the positive a_i's with $1/2(2^\varepsilon - 1)$ and the negative ones with $2/(2^\varepsilon - 1)$), we get that $\frac{1}{2(2^\varepsilon-1)} \sum_{a_j>0} a_j + \frac{2}{(2^\varepsilon-1)} \sum_{a_j<0} a_j < 8m\lambda/(2^\varepsilon - 1)$, which yields that

$$\sum_{a_j>0} a_j + 4 \cdot \sum_{a_j<0} a_j < 16m\lambda. \tag{16}$$

We conclude that:

$$\overrightarrow{\mathsf{W1}}(d, d') = \frac{1}{m} \cdot \sum_{a_j>0} a_j \leq 16\lambda - \frac{4}{m} \cdot \sum_{a_j<0} a_j = 16\lambda + 4 \cdot \overleftarrow{\mathsf{W1}}(d, d').$$

□

Proof of Lemma 4.6

Proof. The Protocol VerifyEmpLabel is defined as follows.

Protocol 4.24.
$\mathsf{VerifyEmpLabel} = (\mathsf{P}_{\mathsf{VEL}}, \mathsf{V}_{\mathsf{VEL}})$.

Common input: An efficiently decidable set $\mathcal{S} \subseteq \{0,1\}^n$, an efficiently computable function $f : \{0,1\}^n \to \{0,1\}^*$, a claimed size s for $\mathcal{S}$, the histogram parameter ε, a claimed histogram h of f over $\mathcal{S}$, the empirical samples $x_1, \ldots, x_\ell$, and claimed bins for the empirical samples u.

Description:

1. $\mathsf{V}_{\mathsf{VEL}}$ *verifies that* $x_1, \ldots, x_\ell \in \mathcal{S}$.
2. *(Preimage tests) The parties interact for every* $i \in [\ell]$*, in parallel, in an execution of the Set Lower-bound protocol of Lemma 3.16, on input* $(f^{-1}(f(x_i)), s \cdot 2^{-\varepsilon\cdot(u(i)+2)}, \varepsilon)$*, where* $\mathsf{P}_{\mathsf{VEL}}$ *and* $\mathsf{V}_{\mathsf{VEL}}$ *play the role of* $\mathsf{P}^i_{\mathsf{LB}}$ *and* $\mathsf{V}^i_{\mathsf{LB}}$ *respectively.*
3. *Let* $h^u = \mathsf{Hist}(u)$ *(see Definition 3.3). The verifier rejects if* $\mathsf{W1}(h, h^u) > \varepsilon$ *(and accepts if not rejected so far).*

Let SetLB^j be the j'th execution of the set Lower-bound protocol in a random execution of VerifyEmpLabel. In the following we prove the completeness and soundness properties of Protocol 4.24.

Completeness. Suppose that the prover is honest (namely $u = u^f$ and $h = h^f$), it follows that

- $\mathsf{V}_{\mathsf{VEL}}$ always accepts in Step 1.
- Proposition 3.4 yields that $|f^{-1}(f(x_i))| \geq |\mathcal{S}| \cdot 2^{-\varepsilon\cdot(u(i)+1)}$, and the promise on s yields that $|\mathcal{S}| \geq 2^{-\varepsilon}s$. Therefore we have $|f^{-1}(f(x_i))| \geq s \cdot 2^{-\varepsilon\cdot(u(i)+2)}$ and by the completeness of Lemma 3.16 $\mathsf{V}^i_{\mathsf{LB}}$ accepts with probability at least $1-2^{-n}$. Hence $\mathsf{V}_{\mathsf{VEL}}$ does not reject in Step 2 with probability at least $1-m2^{-n}$.
- $\mathsf{V}_{\mathsf{VEL}}$ always accepts in Step 3.

Therefore $\mathsf{V}_{\mathsf{VEL}}$ accepts with probability at least $1-m2^{-n} > 1-2^{-n/2}$.

Soundness. We claim that $u^f(i) \leq u(i)+3$ for all $i \in [\ell]$ or otherwise $\mathsf{V}_{\mathsf{VEL}}$ rejects with probability at least $1-2^{-n}$. Let assume that $u^f(i) \geq u(i)+4$ for some $i \in [\ell]$. Then Proposition 3.4 yields that $\left|f^{-1}(f(x_i))\right| \leq |\mathcal{S}| \cdot 2^{-\varepsilon\cdot u^f(i)} \leq |\mathcal{S}| \cdot 2^{-\varepsilon\cdot(u(i)+4)}$. Now since $|\mathcal{S}| \leq s \cdot 2^{\varepsilon}$, it follows that

$$\left|f^{-1}(f(x_i))\right| \leq 2^{-\varepsilon}(s \cdot 2^{-\varepsilon\cdot(u(i)+2)}) \tag{17}$$

Hence, by the soundness of Lemma 3.16, $\mathsf{V}^i_{\mathsf{LB}}$ (and thus $\mathsf{V}_{\mathsf{SWS}}$) rejects with probability least $1-2^{-n}$, in which case we are done.

So in the following we assume that $u^f(i) \leq u(i)+3$ for all $i \in [\ell]$, which by Proposition 3.11 yields that

$$\overleftarrow{\mathsf{SH}}(u^f, u) \leq 3/m. \tag{18}$$

Using the promise $\mathsf{W1}(h^{u^f}, h^f) \leq \varepsilon$ and that $\mathsf{W1}(h^u, h) \leq \varepsilon$ (since otherwise $\mathsf{V}_{\mathsf{VEL}}$ would reject in Step 3), we conclude that

$$\begin{aligned}
&\mathsf{SH}(u^f, u) && (19)\\
&= \overleftarrow{\mathsf{SH}}(u^f, u) + \overrightarrow{\mathsf{SH}}(u^f, u) && \text{Definition 3.10}\\
&\leq \overleftarrow{\mathsf{SH}}(u^f, u) + (\overleftarrow{\mathsf{SH}}(u^f, u) + \overrightarrow{\mathsf{W1}}(h^{u^f}, h^u)) && \text{Lemma 3.12}\\
&\leq 6/m + \overrightarrow{\mathsf{W1}}(h^{u^f}, h^f) + \overrightarrow{\mathsf{W1}}(h^f, h) + \overrightarrow{\mathsf{W1}}(h, h^u) && \text{Equation 18 and Proposition 3.9}\\
&\leq 6/m + \mathsf{W1}(h^{u^f}, h^f) + \overrightarrow{\mathsf{W1}}(h^f, h) + \mathsf{W1}(h, h^u) && \text{Definition 3.8}\\
&\leq 6/m + \varepsilon + \overrightarrow{\mathsf{W1}}(h^f, h) + \varepsilon && \text{Promise and Step 3}\\
&\leq 6/m + 2\varepsilon + (100\varepsilon + 4\overleftarrow{\mathsf{W1}}(h^f, h)) && \text{Promise}\\
&\leq 6/m + 102\varepsilon + 4(\overleftarrow{\mathsf{W1}}(h^f, h^{u^f}) + \overleftarrow{\mathsf{W1}}(h^{u^f}, h^u) + \overleftarrow{\mathsf{W1}}(h^u, h)) && \text{Proposition 3.9}\\
&\leq 6/m + 102\varepsilon + 4(\mathsf{W1}(h^f, h^{u^f}) + \overleftarrow{\mathsf{SH}}(u^f, u) + \mathsf{W1}(h^u, h))) && \text{Lemma 3.12 and Definition 3.8}\\
&\leq 6/m + 102\varepsilon + 4(\varepsilon + 3/m + \varepsilon) && \text{Step 1, Equation 18 and Step 3}\\
&\leq 18/m + 110\varepsilon \leq 111\varepsilon. && \text{for } m\varepsilon = n \geq 18\\
& && (20)
\end{aligned}$$

□

5 Applications

In this section we use the sample with size protocol described in Section 4 for proving our main result. We first formally defined the interactive sampler, Sam, inspired by the sampler of Haitner et al. [32].

Definition 5.1 (Sam). *For $d \in \mathbb{N}$, the randomized stateful oracle Sam_d is defined as follows: on input $(C_1, \ldots, C_i, x)$, where $x \in \{0,1\}^m$ and each C_j is a circuit over $\{0,1\}^m$,*

1. *Sam_d returns $\perp$ if either*
   - *$i > d$, or*
   - *it was not previously asked on $(C_1, \ldots, C_{i-1}, x')$ (for some $x' \in \{0,1\}^m$) and answered with x.*
2. *Otherwise, Sam_d returns a random element in $\mathcal{S}(C_1, \ldots, C_{i-1}, x) := \{x' \in \{0,1\}^m : \forall j \in (i-1) \colon C_j(x') = C_j(x)\}$, where Sam_d uses fresh randomness for each call.*

*Given an oracle-aided (random) algorithm A and $x \in \{0,1\}^*$, we let $A^{\mathsf{Sam}_d}(x)$ be the output distribution of A^{Sam_d} on input x (this distribution is induced by the random coins of A and Sam_d). We say that $A^{\mathsf{Sam}_d}(x)$ is k-*adaptive*, for $k \in \mathbb{N}$, if $A(x)$ makes at most k* parallel *calls to Sam_d, where a parallel call consist of arbitrary many different inputs to Sam_d (i.e., , $(q_1 = C_{1,1}, \ldots, C_{1,j_1}, x_1), \ldots, q_t = (C_{t,1}, \ldots, C_{t,j_t}, x_t))$).*

Given the above definition, we can formally state our main result.

Theorem 5.2 (Restating Theorem 1.1). *For any $d = O(1)$, let A be an efficient oracle-aided algorithm and let $x \in \{0,1\}^*$. If $A^{\mathsf{Sam}_d}(x)$ is k-adaptive, then there exists an $\mathbf{AM}[O(k)]$ protocol AM–$\mathsf{Sam} = (\mathsf{P}, \mathsf{V})$ whose parties get as input $x \in \{0,1\}^*$ and an accuracy parameter $\delta > 1/\operatorname{poly}(|x|)$, the prover P is in $\in \mathbf{BPP}^{\mathbf{NP}}$, and the following hold:*

Completeness: *V accepts in $\langle \mathsf{P}, \mathsf{V} \rangle(\delta, x)$ with probability at least $1 - \delta$.*

Soundness: *For every prover P^* it holds that*

$$\Delta(A^{\mathsf{Sam}_d}(x), \langle \mathsf{P}^*, \mathsf{V} \rangle_{\mathsf{V}}(\delta, x)) \leq \Pr[\langle \mathsf{P}^*, \mathsf{V} \rangle_{\mathsf{V}}(\delta, x) = \perp] + \delta,$$

where $A^{\mathsf{Sam}_d}(x)$ denotes the output of A on input x, and $\langle \mathsf{P}^*, \mathsf{V} \rangle_{\mathsf{V}}(\delta, x)$ denotes the output of V at the end of the interaction with P^* on input (δ, x) (equals $\perp$ if V aborts).

The above theorem yields the following classification.

Corollary 5.3. *Let A be a k-adaptive efficient oracle-aided algorithm such that A^{Sam_d} decides a language $L \subseteq \{0,1\}^n$ — for every $x \in \{0,1\}^n$ it holds that $\Pr[A^{\mathsf{Sam}_d}(x) = 1_L(x)] \geq \frac{1}{2} + \delta$ for $\delta > 1/\operatorname{poly}(n)$. Then $L \in \mathbf{AM}[k] \cap \mathbf{coAM}[k]$, with provers in $\mathbf{BPP}^{\mathbf{NP}}$.*

Proof. In order to keep the text simple, we assume that A makes no parallel queries. Let $\ell < \operatorname{poly}(|x|)$ be an upper bound on the running A on inputs of length $|x|$. We consider the following protocol for the emulation of $A^{\mathsf{Sam}_d}(x)$:

Protocol 5.4.
AM–$\mathsf{Sam} = (\mathsf{P}, \mathsf{V})$.

Common input: An accuracy parameter δ and $x \in \{0,1\}^*$.

Description: For $i \in [d]$ let $\delta_i = (\delta_{i+1})^c/c \cdot \ell^8$, where $\delta_d = \delta/2\ell$ and c is the constant stated in Corollary 4.3.[9]

1. V *chooses, uniformly at random, random coins for A, and initialized a table* Prefix *(initially empty).*
2. V *emulates $A^{\mathsf{Sam}_d}(x)$, while doing the following each time A makes a query $q = (C_1, \ldots, C_i, x)$ to* Sam_d*:*
   (a) *If $i > d$, or $i > 1$ and $\mathsf{st} = (C_1, \ldots, C_{i-1}, x) \notin \mathsf{Prefix}$, then V returns $\perp$ to A as the answer of* Sam_d*.*
   (b) *Otherwise,* P *and* V *are engaged in a random execution of protocol* $\mathsf{SampleWithSize}$ *from Corollary 4.3 on input $(\delta_i, \mathcal{S}(\mathsf{st}), \mathsf{Prefix}(\mathsf{st}), C_i)$,* [10] *where V and P act as the verifier and prover respectively.*
   Let (x', s) be the output of the verifier in the above execution, V stores $\mathsf{Prefix}(C_1, \ldots, C_i, x') = s$ and returns x' to A as the answer of Sam_d*.*
3. V *rejects if it has rejected in one of the above executions.*

It is clear that the complexity of the above protocol matches the statement of the theorem (recall that the prover in $\mathsf{SampleWithSize}$ is in $\mathbf{BPP}^{\mathbf{NP}}$). We will prove the completeness and soundness of the protocol using induction. In the following we assume that V never sets $\mathsf{Prefix}(C_1, \ldots, C_i, x) = s$ for $s \notin [(1 \pm \delta_i) \cdot |\mathcal{S}(C_1, \ldots, C_{i-1}, x)|]$ (i.e., we assume a variant of V that aborts if the original verifier is about to store an invalid value). Corollary 4.3 yields that by doing that we increase the rejecting probability of V by at most $\delta/2$.

Let $\mathrm{View}^j_{\mathsf{Sam}_d}$ denote the view of A after the j'th query to Sam_d. For a prover P^*, we let $\mathrm{View}^j_{\mathsf{P}^*}$ denote the view of A after the j'th query in the emulation done in $\langle \mathsf{P}^*, \mathsf{V} \rangle(\delta, x)$ (where we set it to $\perp$ is V has rejected). Assume that the following for $j \in (\ell)$:

Completeness: V rejects with probability at most $j\delta/2\ell$ when interacting with P up until and including the j'th emulated query.

Soundness: For any (unbounded) prover P^* it holds that $\Delta(\mathrm{View}^j_{\mathsf{Sam}_d}, \mathrm{View}^j_{\mathsf{P}^*}) \leq \rho_j + j\delta/2\ell$, where ρ_j is the probability that V rejects in the first j queries of $\langle \mathsf{P}^*, \mathsf{V} \rangle(\delta, x)$.

Since the output of the verifier at the end of the emulation is a function of A's view, the above for $j = \ell$ yields the proof of the lemma. For proving the case $j+1$, fix any non-rejecting view v for the first j steps of A. Since ℓ bounds the domain of the set parameter $\mathcal{S}$ in any query made by $A(x)$, Corollary 4.3 yields that the following with respect to the $j+1$ query of $A(x)$:

1. V reject with probability at most $\delta/2\ell$ when interacting with P, and

[9] Since d is constant, all these values are inverse polynomials of $|x|$ and $1/\delta$.
[10] Where we view a circuit C with m input wires, as a function over $\{0,1\}^m$.

2. $\Delta((\text{View}^{j+1}_{\mathsf{Sam}_d} \mid v), (\text{View}^{j+1}_{\mathsf{P}^*} \mid v)) \leq \rho_v + \delta/2\ell$, where ρ_v is the probability that V reject in the $j+1$ query, conditioned on v.

The completeness for the $(j+1)$ step follows immediately from the above and the induction hypothesis. For the soundness, note that $\rho_{j+1} = \rho_j + (1-\rho_j) \cdot \mathrm{E}_{v \leftarrow \text{View}^j_{\mathsf{P}^*}}[\rho_v \mid v \neq \perp]$. Similarly, the triangle inequality yields that $\Delta(\text{View}^{j+1}_{\mathsf{Sam}_d}, \text{View}^{j+1}_{\mathsf{P}^*}) \leq \Delta(\text{View}^{j}_{\mathsf{Sam}_d}, \text{View}^{j}_{\mathsf{P}^*}) + (1-\rho_j) \cdot \mathrm{E}_{v \leftarrow \text{View}^j_{\mathsf{P}^*}}[\Delta((\text{View}^{j+1}_{\mathsf{Sam}_d} \mid v), (\text{View}^{j+1}_{\mathsf{P}^*} \mid v)) \mid v \neq \perp] \leq \rho_{j+1} + (j+1)\delta/2\ell$. □

5.1 Lower Bounds on Statistically Hiding Commitments

In this section we use Theorem 5.2 to derive a lower bound on the possibility of basing constant-round statistically hiding commitments on the assumption that $\mathbf{P} \neq \mathbf{NP}$. Statistically hiding commitments are defined in Section 5.1.1. In Section 5.1.2 we show that Sam_d can be used to break any d rounds statistically hiding commitment, and define a reduction from statistically hiding commitments to deciding a language in Section 5.1.2. Finally, we state and prove the result of this section in Section 5.1.4.

5.1.1 Statistically hiding commitments

Definition 5.5 (Statistically hiding commitments)**.** *A* (bit) commitment scheme $(\text{Send}, \text{Rec})$ *is an efficient two-party protocol consisting of two stages.*[11] *Throughout, both parties receive the security parameter* 1^n *as input.*

> **Commit**. *The sender* Send *has a private input* $b \in \{0,1\}$*, which she wishes to commit to the receiver* Rec*, and a sequence of coin tosses* r*. At the end of this stage, both parties receive as common output a* commitment z*.*
>
> **Reveal**. *Both parties receive as input a commitment* z*.* Send *also receives the private input* b *and coin tosses* r *used in the commit stage. This stage is non-interactive:* Send *sends a single message to* Rec*, and* Rec *either outputs a bit (and accepts) or rejects.*

Definition 5.6. *A commitment scheme* $(\text{Send}, \text{Rec})$ *is* statistically hiding *if*

> **Completeness**. *If both parties are honest, then for any bit* $b \in \{0,1\}$ *that* Send *gets as private input,* Rec *accepts and outputs* b *at the end of the reveal stage.*
>
> **Statistical Hiding**. *For every unbounded strategy* Rec^**, the distributions* $\text{View}_{\text{Rec}^*}(\text{Send}(0), \text{Rec}^*)$ *and* $\text{View}_{\text{Rec}^*}(\text{Send}(1), \text{Rec}^*)$ *are statistically indistinguishable.*
>
> **Computational Binding**. *For every* PPT Send^**,* Send^* *succeeds in the following game (breaks the commitment) with negligible probability in* n*:*
>
> - Send^* *interacts with an honest* Rec *in the commit stage, which yields a commitment* z*.*
> - Send^* *outputs two messages* τ_0, τ_1 *such that for both* $b = 0$ *and* $b = 1$*,* Rec *on input* (z, τ_b) *accepts and outputs* b*.*

[11] Sice we are interested in lower bounds, we only present the definition for bit commitments.

5.1.2 Sam and statistically hiding commitments

Haitner et al. [32] (following Wee [63]) showed that Sam can be used for breaking any statistically hiding commitment. Since there are slight differences between the definition of Sam considered above and the one considered in [32], we restate their result according to our formulation and sketch its proof.

Lemma 5.7 (implicit in [32]). *For any d-round statistically hiding commitment* $(\text{Send}, \text{Rec})$*, there exists a deterministic oracle-aided adversary A such that A^{Sam_d} break the binding of* $(\text{Send}, \text{Rec})$ *with save but negligible probability.*

Proof Sketch. We assume without loss of generality that Rec speaks first, and let m be the number of random coins used by Send. We also assume that Send gets its random coins, r, as an additional input (i.e., we view Send's input as a pair $x = (b, r)$, where b is the secret bit of Send).

Let $x_0 := \perp$. In the commit stage, A behaves as follows: given a query q_i from Rec, it queries Sam_d on (C_i, x_{i-1}) to get an answer x_i, where C_i is the following circuit: on input $x \in \{0,1\}^{m+1}$, it outputs the message of $\text{Send}(x)$ on the i'th round, given that Rec's first i messages are $q_1, \ldots, q_i$. Finally, A sends $C_i(x_i)$ to Rec (as its i'th message).

In the reveal stage, A queries Sam_d n times on (C', x_d), where C' is an arbitrary circuit over $\{0,1\}^n$, to get outputs $\{(b_i, r_i)\}_{i=1,\ldots,n}$. If there exists $i \neq j$ such that $b_i \neq b_j$ then A outputs $((b_i, r_i), (b_j, r_j))$, otherwise A aborts.

The definition of Sam_d yields that each (b_i, r_i) is a *random* valid decommitment. Hence, the statistically hiding property of $(\text{Send}, \text{Rec})$ yields that, with save but negligible probability, there exist $b_i \neq b_j$ and A successfully produces decommitments to both 0 and 1.[12] Therefore, A^{Sam_d} breaks the binding of $(\text{Send}, \text{Rec})$ with save but negligible probability.

□

5.1.3 Black-box reductions

We now formally define the notion of black-box reductions from deciding a language to (breaking the binding of) commitment schemes.

Definition 5.8 (Black-box reduction). *A black-box reduction from deciding a language L to breaking the binding of a commitment protocol* $(\text{Send}, \text{Rec})$ *is an oracle-aided algorithm* $(\text{Send}, \text{Rec})$ *with the following guarantee: given as oracle a* deterministic *and* stateless *adversary $\mathcal{O}$ that breaks the binding of* $(\text{Send}, \text{Rec})$*, $R^{\mathcal{O}}$ decides L (i.e., , $\Pr[R^{\mathcal{O}}(x) = 1_L(x)] \geq 1 - 2^{-n}$). We say that R is k-adaptive if it makes k adaptive rounds of queries to its oracle; each round may consist of many queries, but all of the queries in one round can be computed without looking at the oracle responses to any of the other queries in the same round.*

5.1.4 On basing statistically hiding commitments on NP-harndess

Given the above definitions, we can formally state result about reducing the security of statistically hiding commitment on the decidability hardness of a given language.

[12] The statistically hiding property yields that given a random transcript of the commitment, essentially half of Send' possible input pairs that are consistent with the transcript are of the form $(0, \cdot)$, and the other half are of the form $(1, \cdot)$.

Corollary 5.9. *Suppose that there exists an efficient k-adaptive black-box reduction R from deciding a language L to breaking the binding of a statistically hiding commitment. Then $L \in \mathbf{AM}[k] \cap \mathbf{coAM}[k]$ with provers in $\mathbf{BPP}^{\mathbf{NP}}$.*

Proof. Let R and $(\mathrm{Send}, \mathrm{Rec})$ be the assumed reduction and statistically hiding commitment respectively. Let A^{Sam_d} be the algorithm guaranteed by Lemma 5.7 for breaking the binding of $(\mathrm{Send}, \mathrm{Rec})$. We would like to argue that $R^{A^{\mathsf{Sam}_d}}$ decides L, but the problem is that A^{Sam_d} is randomized and stateful. Nevertheless, the following lemma readily follows from Haitner et al. [32].

Lemma 5.10 (implicit in [32])**.** *Let $(\mathrm{Send}, \mathrm{Rec})$, R and L be as in Definition 5.8. The there exists an efficient oracle-aided algorithm $\widetilde{R}$ such that $\widetilde{R}^{\widetilde{\mathcal{O}}}$ decides L for any randomized and stateful oracle $\widetilde{\mathcal{O}}$, which breaks the binding of $(\mathrm{Send}, \mathrm{Rec})$ with save but negligible probability.*

Proof Sketch. We present an efficient algorithm $\widetilde{R}$ and a family of deterministic and stateless oracles $\{\mathcal{O}_\lambda\}$ such that the following hold: 1. with save but negligible probability over the choice of λ, it holds that $\mathcal{O}_\lambda$ breaks the binding of $(\mathrm{Send}, \mathrm{Rec})$, and 2. the execution of $\widetilde{R}^{\widetilde{\mathcal{O}}}(x)$ and $R^{\mathcal{O}_\lambda}$ are statistically close, over the randomness of $\widetilde{R}$, R, $\widetilde{\mathcal{O}}$ and a random choice of λ. Showing that will conclude the proof, since the grantees about $\{\mathcal{O}_\lambda\}$ yields that $R^{\mathcal{O}_\lambda}$ decides L correctly for most λ's, and theretofore $\widetilde{R}^{\widetilde{\mathcal{O}}}$ decided L.

Following [32], we first consider a stateless version $\widehat{\mathcal{O}}$ of $\widetilde{\mathcal{O}}$ that lets the caller hold its state — on each query, the caller provides $\mathcal{O}$ with a state parameter (encoded as string), where at the end of the call $\mathcal{O}$ returns the updated state to the caller (in addition to its original output). We would like to claim that whatever can be done with $\widehat{\mathcal{O}}$ could be done with $\widetilde{\mathcal{O}}$. The problem is, however, that a "user" of $\widehat{\mathcal{O}}$ can get additional power by providing fake states. Following [32], this problem is solved by letting $\widehat{\mathcal{O}}$ sign its states using information theoretic signature (i.e., the output of a random function that $\widehat{\mathcal{O}}$ keeps in its belly), and verify the validity of the signature in each call. Finally, we let $\mathcal{O}_\lambda$ be the oracle $\widehat{\mathcal{O}}$ whose random coins (including the one used for the signatures) fixed to λ.[13]

Since $\widehat{\mathcal{O}}$ breaks the binding of $(\mathrm{Send}, \mathrm{Rec})$ with save but negligible probability, item 1 holds with respect to $\{\mathcal{O}_\lambda\}$. Moreover, the signature mechanism we employ, tell us that, with save but negligible probability over the choice of λ and the random coins of R, invalid calls made by R (i.e., with fake states) are answered with $\perp$.

On input x algorithm $\widetilde{R}$ emulates $R^{\mathcal{O}_\lambda}(x)$ as follows: it forwards the oracle calls of R to $\widetilde{\mathcal{O}}$ (stripped from the state parameter), and returns $\widetilde{\mathcal{O}}$ answers to R, along with the state of $\widetilde{\mathcal{O}}$ and a signature of the state ($\widetilde{R}$ computes both parameters by itself, where for the signature it simply returns a random string). In addition, if the state given in the call is invalid (was not return by a previous call), $\widetilde{R}$ returns $\perp$ as the answer to the call.[14] Finally, it answers identical queries with identical answers (as a stateless oracle should do).

Assuming that R never gets non $\perp$ answers to invalid queries, the distribution of $\widetilde{R}^{\widetilde{\mathcal{O}}}(x)$ and $R^{\mathcal{O}_\lambda}$ are identical. Thus, item 2 follows by the above observation about the guarantee of the signature mechanism.

□

[13] Since the running time of R is bounded, the size of λ is bounded as well.

[14] Note that $\widetilde{R}$ does not need to use the signature mechanism to ensure validity, since $\widetilde{R}$ is stateful and can keep track on the execution.

Lemma 5.10 yields that $\widetilde{R}^{A^{\mathsf{Sam}_d}}$ decides L. Since A is efficient, it follows that there exists an efficient oracle-aided algorithm R' such that R'^{Sam_d} decides L. Hence, Corollary 5.3 yields that $L \in \mathbf{AM}[k] \cap \mathbf{coAM}[k]$ with provers in $\mathbf{BPP}^{\mathbf{NP}}$. □

Acknowledgment

The authors would like to thank Boaz Barak, Thomas Holenstein and Salil Vadhan for very useful discussions. The third author also thanks the other authors of [31] for fruitful discussions and their perspectives on the power of Sam.

References

[1] W. Aiello and J. Håstad. Statistical zero-knowledge languages can be recognized in two rounds. *Journal of Computer and System Sciences*, 42(3):327–345, 1991. Preliminary version in *FOCS'87*.

[2] M. Ajtai. The worst-case behavior of schnorr's algorithm approximating the shortest nonzero vector in a lattice. In *STOC '03*, pages 396–406, New York, NY, USA, 2003. ACM. ISBN 1-58113-674-9. doi: http://doi.acm.org/10.1145/780542.780602.

[3] A. Akavia, O. Goldreich, S. Goldwasser, and D. Moshkovitz. On basing one-way functions on np-hardness. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC)*, pages 701–710, 2006.

[4] N. Alon, L. Babai, and A. Itai. A fast and simple randomized parallel algorithm for the maximal independent set problem. *J. Algorithms*, 7(4):567–583, 1986. ISSN 0196-6774. doi: http://dx.doi.org/10.1016/0196-6774(86)90019-2.

[5] Babai, Fortnow, and Lund. Non-deterministic exponential time has two-prover interactive protocols. *CMPCMPL: Computational Complexity*, 1, 1991.

[6] L. Babai and S. Moran. Arthur-Merlin games: A randomized proof system and a hierarchy of complexity classes. *Journal of Computer and System Sciences*, 36:254–276, 1988.

[7] B. Barak. How to go beyond the black-box simulation barrier. In *Proceedings of the 42nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 106–115, 2001.

[8] M. Blum. How to prove a theorem so no one else can claim it. In *Proceedings of the International Congress of Mathematicians*, pages 1444–1451, 1987.

[9] A. Bogdanov and L. Trevisan. On worst-case to average-case reductions for np problems. *SIAM Journal on Computing*, 36(4):1119–1159, 2006.

[10] A. Bogdanov and L. Trevisan. Average-case complexity. *CoRR*, 2006.

[11] R. B. Boppana, J. Håstad, and S. Zachos. Does co-NP have short interactive proofs? *Information Processing Letters*, 25(2):127–132, 1987.

[12] G. Brassard. Relativized cryptography. In *Proceedings of the 20th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 383–391. IEEE Computer Society, 1979.

[13] G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *Journal of Computer and System Sciences*, 37(2):156–189, 1988.

[14] J. Buchmann, J. Loho, and J. Zayer. An implementation of the general number field sieve. In *CRYPTO '93*, pages 159–165, New York, NY, USA, 1994. Springer-Verlag New York, Inc. ISBN 0-387-57766-1.

[15] I. Damgård. Collision free hash functions and public key signature schemes. In *Advances in Cryptology – EUROCRYPT '87*, volume 304 of *Lecture Notes in Computer Science*, pages 203–216. Springer, 1987.

[16] I. Damgård, O. Goldreich, T. Okamoto, and A. Wigderson. Honest verifier vs. dishonest verifier in public coin zero-knowledge proofs. In *Advances in Cryptology – CRYPTO '95*, volume 963 of *Lecture Notes in Computer Science*, pages 325–338. Springer, 1995.

[17] I. B. Damgård, T. P. Pedersen, and B. Pfitzmann. Statistical secrecy and multibit commitments. *IEEE Transactions on Information Theory*, 44(3):1143–1151, 1998.

[18] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[19] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. volume 196 of *Lecture Notes in Computer Science*, pages 10–18. Springer, 1984.

[20] J. Feigenbaum and L. Fortnow. Random-self-reducibility of complete sets. *SIAM Journal on Computing*, 22(5):994–1005, 1993.

[21] L. Fortnow. The complexity of perfect zero-knowledge. *Advances in Computing Research: Randomness and Computation*, 5:327–343, 1989.

[22] O. Goldreich and S. Goldwasser. On the possibility of basing cryptography on the assumption that P $\neq$ NP. Theory of Cryptography Library: Record 98-05, February 1998. `http://theory.lcs.mit.edu/~tcryptol`.

[23] O. Goldreich and A. Kahan. How to construct constant-round zero-knowledge proof systems for NP. *Journal of Cryptology*, 9(3):167–190, 1996.

[24] O. Goldreich and S. P. Vadhan. Comparing entropies in statistical zero knowledge with applications to the structure of SZK. In *IEEE Conference on Computational Complexity*, pages 54–73. IEEE Computer Society, 1999.

[25] O. Goldreich, Y. Mansour, and M. Sipser. Interactive proof systems: Provers that never fail and random selection. *Foundations of Computer Science, Annual IEEE Symposium on*, 0:449–461, 1987. ISSN 0272-5428. doi: http://doi.ieeecomputersociety.org/10.1109/SFCS.1987.35.

[26] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the ACM*, 38(1):691–729, 1991. Preliminary version in *FOCS'86*.

[27] O. Goldreich, A. Sahai, and S. Vadhan. Honest-verifier statistical zero-knowledge equals general statistical zero-knowledge. In *In Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 399–408, 1998.

[28] O. Goldreich, S. Vadhan, and A. Wigderson. On interactive proofs with a laconic prover. In *Proc. 28th ICALP*, pages 334–345, 2001.

[29] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.

[30] S. Goldwasser and M. Sipser. Private coins versus public coins in interactive proof systems. *Advances in Computing Research: Randomness and Computation*, 5:73–90, 1989.

[31] S. D. Gordon, H. Wee, A. Yerukhimovich, and D. Xiao. On the round complexity of zero-knowledge proofs from one-way permutations, 2009. Manuscript. Available at http://www.cs.princeton.edu/~dxiao/docs/zk-owp.pdf.

[32] I. Haitner, J. J. Hoch, O. Reingold, and G. Segev. Finding collisions in interactive protocols – A tight lower bound on the round complexity of statistically-hiding commitments. In *Proceedings of the 47th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE Computer Society, 2007.

[33] I. Haitner, M. Nguyen, S. J. Ong, O. Reingold, and S. Vadhan. Statistically-hiding commitments and statistical zero-knowledge arguments from any one-way function. To appear in SIAM Journal on Computing, November 2007.

[34] I. Haitner, O. Reingold, S. Vadhan, and H. Wee. Inaccessible entropy. In *STOC '09: Proceedings of the 41st annual ACM symposium on Theory of computing*, pages 611–620, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-506-2. doi: http://doi.acm.org/10.1145/1536414.1536497.

[35] I. Haitner, A. Rosen, and R. Shaltiel. On the (im)possibility of arthur-merlin witness hiding protocols. In *Theory of Cryptography, Fourth Theory of Cryptography Conference, TCC 2009*, 2009.

[36] Y. Han, L. A. Hemaspaandra, and T. Thierauf. Threshold computation and cryptographic security. *SIAM J. Comput.*, 26(1):59–78, 1997. ISSN 0097-5397. doi: http://dx.doi.org/10.1137/S0097539792240467.

[37] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999. Preliminary versions in *STOC'89* and *STOC'90*.

[38] T. Holenstein. Private communication. 2009.

[39] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography. In *Proceedings of the 30th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 230–235, 1989.

[40] Y. Ishai, E. Kushilevitz, and R. Ostrovsky. Sufficient conditions for collision-resistant hashing. In *In Proceedings of the 2nd Theory of Cryptography Conference*, pages 445–456, 2005.

[41] L. V. Kantorovich. On the translocation of masses. *Doklady Akademii Nauk SSSR*, 37:227–229, 1942.

[42] L. V. Kantorovich and G. S. Rubinstein. On a space of totally additive functions. *Vestn Lening. Univ*, 13(7):52–59, 1958.

[43] Y. Lindell. Parallel coin-tossing and constant-round secure two-party computation. *JCRYPTOLOGY*, 2003.

[44] C. Lund, L. Fortnow, H. Karloff, and N. Nisan. Algebraic methods for interactive proof systems. In *Proc. 31st FOCS*, pages 2–10. IEEE, 1990.

[45] D. Micciancio and O. Regev. Worst-case to average-case reductions based on gaussian measures. In *FOCS '04*, pages 372–381, Washington, DC, USA, 2004. IEEE Computer Society. ISBN 0-7695-2228-9. doi: http://dx.doi.org/10.1109/FOCS.2004.72.

[46] G. Monge. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie des Sciences de Paris*, page 666, 1781.

[47] M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing (STOC)*, pages 33–43. ACM Press, 1989.

[48] M. Naor, R. Ostrovsky, R. Venkatesan, and M. Yung. Perfect zero-knowledge arguments for NP using any one-way permutation. *Journal of Cryptology*, 11(2):87–108, 1998. Preliminary version in *CRYPTO'92*.

[49] T. Okamoto. On relationships between statistical zero-knowledge proofs. *Journal of Computer and System Sciences*, 60(1):47–108, 2000. Preliminary version in *STOC'96*.

[50] S. J. Ong and S. P. Vadhan. An equivalence between zero knowledge and commitments. In R. Canetti, editor, *TCC*, volume 4948 of *Lecture Notes in Computer Science*, pages 482–500. Springer, 2008. ISBN 978-3-540-78523-1.

[51] R. Ostrovsky and A. Wigderson. One-way functions are essential for non-trivial zero-knowledge. In *Proceedings of the 2nd Israel Symposium on Theory of Computing Systems*, pages 3–17. IEEE Computer Society, 1993.

[52] R. Pass. Parallel repetition of zero-knowledge proofs and the possibility of basing cryptography on np-hardness. In *IEEE Conference on Computational Complexity*, pages 96–110, 2006.

[53] A. Pavan, A. L. Selman, S. Sengupta, and N. V. Vinodchandran. Polylogarithmic-round interactive proofs for conp collapse the exponential hierarchy. *Theor. Comput. Sci.*, 385(1-3): 167–178, 2007. ISSN 0304-3975. doi: http://dx.doi.org/10.1016/j.tcs.2007.06.013.

[54] M. O. Rabin. Digitalized signatures and public-key functions as intractable as factorization. Technical Report MIT/LCS/TR-212, Massachusetts Institute of Technology, Jan. 1979. URL ftp://ftp-pubs.lcs.mit.edu/pub/lcs-pubs/tr.outbox/MIT-LCS-TR-212.ps.gz.

[55] O. Reingold, L. Trevisan, and S. P. Vadhan. Notions of reducibility between cryptographic primitives. In *Theory of Cryptography, First Theory of Cryptography Conference, TCC 2004*, volume 2951 of *Lecture Notes in Computer Science*, pages 1–20. Springer, 2004.

[56] R. L. Rivest, A. Shamir, and L. M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, Feb 1978.

[57] Y. Rubner, C. Tomasi, and L. J. Guibas. A metric for distributions with applications to image databases. In *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, page 59, Washington, DC, USA, 1998. IEEE Computer Society. ISBN 81-7319-221-9.

[58] A. Sahai and S. Vadhan. A complete problem for statistical zero knowledge. *Journal of the ACM*, 50(2):196–249, 2003. Preliminary version in *FOCS'97*.

[59] S. Sanghvi and S. P. Vadhan. The round complexity of two-party random selection. In H. N. Gabow and R. Fagin, editors, *STOC*, pages 338–347. ACM, 2005. ISBN 1-58113-960-8.

[60] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26:1484–1509, 1997.

[61] D. Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In *Advances in Cryptology – EUROCRYPT '98*, volume 1403 of *Lecture Notes in Computer Science*, pages 334–345. Springer, 1998.

[62] L. G. Valiant and V. V. Vazirani. Np is as easy as detecting unique solutions. In *STOC '85: Proceedings of the seventeenth annual ACM symposium on Theory of computing*, pages 458–463, New York, NY, USA, 1985. ACM. ISBN 0-89791-151-2. doi: http://doi.acm.org/10.1145/22145.22196.

[63] H. Wee. One-way permutations, interactive hashing and statistically hiding commitments. In *TCC '07*, pages 419–433, 2007.

[64] D. Xiao. (Nearly) optimal black-box constructions of commitments secure against selective opening attacks, 2009. Manuscript.

[65] C.-K. Yap. Some consequences of non-uniform conditions on uniform classes. *Theor. Comput. Sci.*, 26:287–300, 1983.

A Omitted proofs

A.1 Relation between W1 and SH

Proof of Lemma 3.12. Let $J := \left\{ j \in (m) \colon \sum_{i \in (j)} h_i^u > \sum_{i \in (j)} h_i^v \right\}$. By Definition 3.8 it holds that $\overrightarrow{\mathsf{W1}}(h^u, h^v) = \frac{1}{m} \cdot \sum_{j \in J} (\sum_{i \in (j)} h_i^u - \sum_{i \in (j)} h_i^v)$, and so by Definition 3.3 it holds that

$$\sum_{i \in (j)} h_i^u = \frac{1}{\ell} \cdot \sum_{i \in (j)} |\{k \colon u(k) \leq j\}| = \frac{1}{\ell} \cdot (|\{i \colon u(i) \leq j\}| .$$

Now the first part holds since

$$\begin{aligned}
&\overrightarrow{\mathsf{W1}}(h^u, h^v) \\
&= \frac{1}{m\ell} \cdot \sum_{j \in J} (|\{i \colon u(i) \leq j\}| - |\{if \colon v(i) \leq j\}|) \\
&= \frac{1}{m\ell} \cdot \sum_{j \in J} (|\{i \colon u(i) \leq j \wedge v(i) > j\}| - |\{i \colon v(i) \leq j \wedge u(i) > j\}|) \\
&\leq \frac{1}{m\ell} \cdot \sum_{j \in J} (|\{i \colon u(i) \leq j \wedge v(i) > j\}|) \\
&\leq \frac{1}{m\ell} \cdot \sum_{j \in (m)} (|\{i \colon u(i) \leq j \wedge v(i) > j\}|) \\
&= \overrightarrow{\mathsf{SH}}(u, v) \qquad \text{(by Proposition 3.11).}
\end{aligned}$$

Similarly it holds that

$$\begin{aligned}
&\overrightarrow{\mathsf{W1}}(h^u, h^v) \\
&= \frac{1}{m\ell} \cdot \sum_{j \in J} (|\{i \colon u(i) \leq j\}| - |\{i \colon v(i) \leq j\}|) \\
&\leq \frac{1}{m\ell} \cdot \sum_{j \in (m)} (|\{i \colon u(i) \leq j\}| - |\{i \colon v(i) \leq j\}|) \\
&= \frac{1}{m\ell} \cdot \sum_{j \in (m)} (|\{i \colon u(i) \leq j \wedge v(i) > j\}| - |\{i \colon v(i) \leq j \wedge u(i) > j\}|) \\
&= \frac{1}{m\ell} \cdot \Big(\sum_{j \in (m)} |\{i \colon u(i) \leq j \wedge v(i) > j\}| - \sum_{j \in (m)} |\{i \colon v(i) \leq j \wedge u(i) > j\}| \Big) \\
&= \overrightarrow{\mathsf{SH}}(u, v) - \overleftarrow{\mathsf{SH}}(u, v) \qquad \text{(by Proposition 3.11),}
\end{aligned}$$

which proves the second part. □

A.2 Efficient provers for AM protocols

The goal of this section is to prove Lemma 3.15. To do so we start with the following lemma.

Lemma A.1. *Let* $\mathsf{M} = (\mathsf{P}, \mathsf{V})$ *be a* $\mathbf{AM}[O(1)]$ *protocol,* $\delta \geq 1/\operatorname{poly}(n)$*, and the following set is not empty:*

$$\mathcal{Y}_n^{1-\delta} = \{x \in \{0,1\}^n \colon \Pr[\langle \mathsf{P}, \mathsf{V} \rangle(x) \text{ accepts}] \geq 1 - \delta\},$$

then there exists a $\mathbf{BPP}^{\mathbf{NP}}$ *strategy that with probability* $\geq 1 - 2^{-n/2}$ *finds an element* $x \in \{0,1\}^n$ *such that* $\Pr[\langle \mathsf{P}, \mathsf{V} \rangle(x) \text{ accepts}] \geq 1 - 2\delta$*.*

Proof. Define the set $\mathcal{N}_n^{1-2\delta} = \{x \in \{0,1\}^n \colon \Pr[\langle \mathsf{P}, \mathsf{V} \rangle(x) \text{ accepts}] \leq 1 - 2\delta\}$, and let $\mathsf{M}' = (\mathsf{P}', \mathsf{V}')$ be the "amplified" two-round protocol that Lemma 3.14 yields with respect to $\mathsf{M} = (\mathsf{P}, \mathsf{V})$ with parameters $\alpha = 1 - \delta$ and $\beta = 1 - 2\delta$. Thus,

- for every $x \in \mathcal{Y}_n^{1-\delta}$, it holds that $\Pr[\langle \mathsf{P}', \mathsf{V}' \rangle(x) \text{ accepts}] \geq 1 - 2^{-2n}$, and
- for every $x \in \mathcal{N}_n^{1-2\delta}$, it holds that $\Pr[\langle \mathsf{P}', \mathsf{V}' \rangle(x) \text{ accepts}] \leq 2^{-2n}$.

Let ω denote the random coins of V' for inputs of length n. By a union bound, it holds that

$$\Pr_{\omega}[\exists x \in \mathcal{N}_n^{1-2\delta},\ \langle \mathsf{P}', \mathsf{V}' \rangle(x;\omega) \text{ accepts}] \leq 2^{-n} \tag{21}$$

The $\mathbf{BPP}^{\mathbf{NP}}$ algorithm claimed in the theorem simply does the following: choose ω uniformly at random, and use the **NP** oracle to find x such that $\mathsf{V}'(x;\omega) = 1$. Notice that since $\mathcal{Y}_n^{1-\delta} \neq \varnothing$, therefore with probability at least $1 - 2^{-2n}$ such x exists. Furthermore, by Inequality 21 the probability that $x \in \mathcal{N}_n^{1-\delta}$ is at most 2^{-n}.

Therefore, with probability $1 - 2^{-n} - 2^{-2n} \geq 1 - 2^{-n/2}$ we output $x \notin \mathcal{N}_n^{1-\delta}$, namely it will hold that $\Pr[\langle \mathsf{P}, \mathsf{V} \rangle(x) \text{ accepts}] \geq 1 - 2\delta$. □

Proof of Lemma 3.15. For any sequence of i messages $w = (r_1, m_1, r_2, \dots)$ exchanged between V and P we can always define a $\mathbf{AM}[k-i]$ game $(\mathsf{P}^w, \mathsf{V}^w)$ with respect to w where the fist i messages are *fixed* to be w and the parties continue interacting as if they are (P, V). For any sequence of messages w we define $\rho(w) = \Pr[\mathsf{V}^w \text{ accepts in } \langle \mathsf{P}^w, \mathsf{V}^w \rangle]$.

Suppose r_1 is the first message of the verifier V. By an average argument it holds that $\Pr[\rho(r_1) \geq 1 - \sqrt{\delta}] \geq 1 - 2\sqrt{\delta}$. The prover strategy P' pretends that $\rho(r_1) \geq 1 - \sqrt{\delta}$ holds and uses Lemma A.1 where the input x of Lemma A.1 will be the response of P' to the message r_1. Lemma A.1 yields that if $\rho(r_1) \geq 1 - \sqrt{\delta}$ then with probability $1 - \text{neg}(n)$, P' finds a message $x = m_1$ such that $\rho(r_1, m_1) \geq 1 - 2\sqrt{\delta}$. Inductively if $\rho(r_1, m_1, \dots, r_{i-1}, m_{i-1}) \geq 1 - 2\delta^{1/2^i}$, then it holds that

$$\Pr[\rho(r_1, m_1, \dots, r_{i-1}, m_{i-1}, r_i) \geq 1 - \delta^{1/2^{i+1}}] \geq 1 - 2\delta^{1/2^{i+1}}$$

and if $\rho(r_1, m_1, \dots, r_{i-1}, m_{i-1}, r_i) \geq 1 - \delta^{1/2^{i+1}}$, then the prover can use the $\mathbf{BPP}^{\mathbf{NP}}$ strategy of Lemma A.1 to find m_i such that $\rho(r_1, m_1, \dots, r_{i-1}, m_{i-1}, r_i, m_i) \geq 1 - 2\delta^{1/2^{i+1}}$. If at the end P' achieves $\rho(r_1, \dots, m_k) > 0$ he succeeds. By a union bound the latter happens with probability at least $1 - 2\sum_{i \in [k]} \delta^{1/2^i} > 1 - 2k\delta^{1/2^k}$.

□

A.3 Sampling from decidable sets

Proof of Lemma 3.18.

Protocol A.2. *Set $k = \log(5/\delta)$ and $\ell = \left\lfloor \log(\frac{(\delta/5)^3 s}{2k^2}) \right\rfloor$. Let $H_{n,\ell}$ be an efficient family of $2k$-wise independent hash functions mapping n bits to ℓ bits. Set $t = \frac{1-\delta/5}{1+\delta/32} \cdot \frac{s}{2^\ell}$.*

1. V_{US} *picks $h \leftarrow H_{n,\ell}$ and sends h to P_{US}.*
2. P_{US} *computes distinct $x_1, \dots, x_t \in \mathcal{S} \cap h^{-1}(0)$ and sends them to the verifier (or aborts if such $x_1, \dots, x_t$ do not exist).*
3. V_{US} *checks that she has received distinct $x_1, \dots, x_t$ and that $x_i \in \mathcal{S} \cap h^{-1}(0)$ for all $i \in [t]$ and reject if any of the checks does not hold. If not rejected, pick $i \leftarrow [t]$ and output x_i.*

Completeness. The only case where the honest prover might cause the verifier to abort is if there do not exist t distinct elements in $|\mathcal{S} \cap h^{-1}(0)|$, which we call the bad event W. For every $x \in \mathcal{S}$ define ζ_x to be the random variable that is 1 if $h(x) = 0$ and zero otherwise, and let $\overline{\zeta_x} = \zeta_x - 2^{-\ell}$ such that $\mathrm{E}[\overline{\zeta_x}] = 0$. We derive:

$$\begin{aligned}
\Pr[W] &= \Pr[\sum_{x\in\mathcal{S}} \zeta_x < t] \\
&= \Pr[\sum_{x\in\mathcal{S}} \zeta_x < \tfrac{1-\delta/5}{1+\delta/32} \cdot \tfrac{s}{2^\ell}] \\
&\le \Pr[\sum_{x\in\mathcal{S}} \zeta_x < (1-\delta/5)|\mathcal{S}|/2^\ell] && \text{by the promise } s \in [(1 \pm \delta/32)|\mathcal{S}|] \\
&\le \Pr\left[\sum_{x\in\mathcal{S}} \overline{\zeta_x} < -(\delta/5)|\mathcal{S}|2^{-\ell}\right] \\
&\le \frac{\mathrm{E}\left[\left(\sum_{x\in\mathcal{S}} \overline{\zeta_x}\right)^{2k}\right]}{((\delta/5)|\mathcal{S}|2^{-\ell})^{2k}} && \text{raise to } 2k \text{ power and use Markov} \\
&= \frac{\mathrm{E}\left[\sum_{x_1,\ldots x_{2k}\in\mathcal{S}} \prod_{i=1}^{2k} \overline{\zeta_{x_i}}\right]}{(\delta/5)^{2k}|\mathcal{S}|^{2k}2^{-2k\ell}} \\
&= \frac{\sum_{x_1,\ldots x_{2k}\in\mathcal{S}} \mathrm{E}\left[\prod_{i=1}^{2k} \overline{\zeta_{x_i}}\right]}{(\delta/5)^{2k}|\mathcal{S}|^{2k}2^{-2k\ell}}.
\end{aligned}$$

By $2k$-wise independence, all terms in the numerator of the last expression above given by tuples $(x_1, \ldots, x_{2k})$ where one of the x_i is unique will contribute 0 to the sum. Therefore it suffices to count such tuples where no x_i is unique.

Claim A.3.

$$\sum_{x_1,\ldots,x_{2k}\in\mathcal{S}} \mathrm{E}\left[\prod_{i=1}^{2k} \overline{\zeta_{x_i}}\right] \le |\mathcal{S}|^k k^{2k} 2^{-k\ell}.$$

By Claim A.3 we conclude that

$$\Pr[W] \le \frac{|\mathcal{S}|^k k^{2k} 2^{-k\ell}}{(\delta/5)^{2k}|\mathcal{S}|^{2k}2^{-2k\ell}} \le \left(\frac{k^2 2^\ell}{(\delta/5)^2|\mathcal{S}|}\right)^k \le (\delta/10)^k < \delta,$$

as claimed.

Proof. (of Claim A.3) There are $\binom{|\mathcal{S}|}{i}$ ways of choosing i elements out of $|\mathcal{S}|$, and there are at most i^{2k} ways of arranging these elements (with duplicates) when there are $2k$ total elements. For all $c \ge 2$, the expectation $\mathrm{E}[(\overline{\zeta_x})^c] \le 2^{-\ell}$. Therefore, the sum of expectations over all tuples with exactly i non-unique elements is bounded by $\le \binom{|\mathcal{S}|}{i} i^{2k} 2^{-i\ell}$. The entire sum is bounded by $\sum_{i=1}^{k} \binom{|\mathcal{S}|}{i} i^{2k} 2^{-i\ell}$ (we do not sum $i > k$ as these terms have duplicate x_i's and so therefore their expectation is 0). By Stirling's approximation the maximum term is for $i = k$, therefore the sum of all elements is bounded by $k\binom{|\mathcal{S}|}{k} \cdot k^{2k} 2^{-k\ell} \le |\mathcal{S}|^k \cdot k^{2k} 2^{-k\ell}$. □

Soundness. Since a random sample from $\mathcal{S}$ is never the failure symbol $\perp$ while the protocol might output $\perp$, it follows that

$$\begin{aligned}\Delta(x,\mathcal{S}) &= \max_{T\subseteq\mathcal{S}\cup\{\perp\}}\{\Pr[x\in T]-\Pr[\mathcal{S}\in T]\}\\ &= \Pr[x=\perp]+\max_{T\subseteq\mathcal{S}}\{\Pr[x\in T]-\Pr[\mathcal{S}\in T]\}.\end{aligned}$$

Therefore, it suffices (and is actually equivalent) to show that for all $T\subseteq\mathcal{S}$, it holds that

$$\Pr[x\in T]\le\frac{|T|}{|\mathcal{S}|}+\delta.$$

Now fix any $T\subseteq\mathcal{S}$, and if $|T|<(\delta/4)|\mathcal{S}|$ then simply pad T with extra elements until we have $|T|=(\delta/4)|S|$. Let W_T be the event that $|h^{-1}(0)\cap T|>(1+\delta/5)2^{-\ell}|T|$ elements. Let $\zeta_x,\overline{\zeta_x}$ be defined as in the case of completeness.

$$\begin{aligned}\Pr[W_T] &= \Pr[\sum_{x\in T}\zeta_x>(1+\delta/5)2^{-\ell}|T|]\\ &= \Pr[\sum_{x\in T}\overline{\zeta_x}>(\delta/5)2^{-\ell}|T|]\\ &\le \frac{\sum_{x_1,\ldots,x_{2k}}\mathrm{E}[\prod_{i=1}^{2k}\overline{\zeta_{x_i}}]}{(\delta/5)^{2k}2^{-2k\ell}|T|^{2k}}\\ &\le \left(\frac{k^2 2^\ell}{(\delta/5)^2|T|}\right)^k\end{aligned}$$

Using the assumption $|T|\ge(\delta/5)|\mathcal{S}|$ and the definition of ℓ,k, we have that $\frac{k^2 2^\ell}{(\delta/5)^2|T|}\le 1/2$ and so the probability that W_T occurs is $\le\delta/5$. Assuming W_T did not occur, we have that

$$\begin{aligned}|T\cap h^{-1}(0)| &\le (1+\delta/5)2^{-\ell}|T|\\ &= \frac{(1+\delta/5)(1+\delta/32)}{1-\delta/5}\cdot\frac{|T|}{s}\cdot t\\ &\le \frac{(1+\delta/5)(1+\delta/32)}{(1-\delta/5)(1-\delta/32)}\cdot\frac{|T|}{|\mathcal{S}|}\cdot t\\ &\le (1+\delta/2)\cdot\frac{|T|}{|\mathcal{S}|}\cdot t.\end{aligned}$$

Therefore the probability of sampling an element of T is $\delta/4+(1+\delta/2)\frac{|T|}{|\mathcal{S}|}$, where the first term is the probability of the bad event W_T occuring while the second is the probability that conditioned on the bad event not occurring, we sample an element of T. If T was not padded then it holds that

$$\delta/4+(1+\delta/2)\tfrac{|T|}{|\mathcal{S}|}\le\tfrac{|T|}{|\mathcal{S}|}+\delta/2(1/2+\tfrac{|T|}{|\mathcal{S}|})\le\tfrac{|T|}{|\mathcal{S}|}+3\delta/4.$$

In case T was padded, we again get that the probability is at most

$$\delta/4+(1+\delta/2)\delta/4\le 3\delta/4\le\tfrac{|T|}{|\mathcal{S}|}+3\delta/4$$

□

B The Case of Sam_2

In this section we present a simple proof of Theorem 5.2 for the special case of $d = 2$ (i.e., non-recursive collision finder). In this simplified proof, the following upper-bound protocol takes the role of the more complex Protocol 4.7.

Lemma B.1 (Set Upper-bound protocol[1])**.** *There exists a two-round public-coin protocol* $\mathsf{SetUB} = (\mathsf{P}_{\mathsf{SUB}}, \mathsf{V}_{\mathsf{SUB}})$*, where the parties get as input an efficiently decidable set* $\mathcal{S} \subseteq \{0,1\}^n$*,* s *(as size of* $\mathcal{S}$*),* $\delta, \varepsilon\, 0$*, the verifier gets in addition a* secret *random sample* x *from* $\mathcal{S}$ *(unknown to the prover) and runs in time* $\mathrm{poly}(n, 1/\delta, 1/\varepsilon)$*, and the following hold:*

Completeness. *If* $|\mathcal{S}| \leq s$*, then* $\Pr[\mathsf{V}_{\mathsf{SUB}}$ *accepts in* $\langle \mathsf{P}_{\mathsf{SUB}}, \mathsf{V}_{\mathsf{SUB}}(x)\rangle(\mathcal{S}, s, \delta, \varepsilon)] \geq 1 - \delta$*.*

Soundness. *If* $|\mathcal{S}| \geq s(1+\varepsilon)$*, then for every prover* P^**, it holds that* $\Pr[\mathsf{V}_{\mathsf{SUB}}$ *accepts in* $\langle \mathsf{P}^*, \mathsf{V}_{\mathsf{SUB}}\rangle(\mathcal{S}, s, \delta, \varepsilon)] < 1 - \varepsilon/5 + \delta$*.*

Theorem B.2. $\mathbf{BPP}^{\mathsf{Sam}_2[k]} \subseteq \mathbf{AM}[k] \cap \mathbf{coAM}[k]$*.*

Proof Sketch.(of Theorem 5.2 for the case $d = 2$) This sketch is very high-level as it is meant only to illustrate the idea and to show that the case of Sam_2 is simpler than the case of Sam_d for $d > 2$; a formal proof of this theorem is left as a special case of our general Theorem 5.2.

Given an efficient oracle aided algorithm A, we construct an **AM** protocol that emulates A^{Sam_2} as follows: The protocol's high-level strategy is standard: the verifier tries to emulate the execution of A^{Sam_2} by picking random coins for the reduction A, and whenever A asks an oracle query to Sam_2, the verifier engages the prover in a protocol such that the distribution of the output is close to what Sam_2 would output, or else if not the verifier rejects.

Depth 1 queries: a query $(C_1, \perp)$ is answered as follows. Setting $\varepsilon = 1/\mathrm{poly}(n)$ suitably small, $\ell = (1/\varepsilon^2)$ and $\delta = \varepsilon^8$, the verifier runs

1. $\mathsf{V}_{\mathsf{SUB}}$ samples $x_1, \ldots, x_\ell \leftarrow \{0,1\}^n$ at random and sends all the $y_i = C_1(x_i)$ to $\mathsf{P}_{\mathsf{SUB}}$.
2. $\mathsf{P}_{\mathsf{SUB}}$ responds with $s_i = |C_1^{-1}(C_1(x_i))|$ for each $i \in [\ell]$.
3. for each $i \in [\ell]$ in parallel:
   (a) Using the lower-bound protocol of Lemma 3.16 on input $(\mathcal{S}, \varepsilon, \delta)$, $\mathsf{V}_{\mathsf{SUB}}$ verifies that $s_i \leq |C_1^{-1}(C_1(x_i))|$.
   (b) Using the upper-bound protocol of Lemma B.1 on common input $(\mathcal{S}, \delta, \varepsilon)$ and secret input x_i, $\mathsf{V}_{\mathsf{SUB}}$ verifies that $s_i \geq |C_1^{-1}(C_1(x_i))|$.
4. $\mathsf{V}_{\mathsf{SUB}}$ rejects if one of the verifier in one of the above execution does. Otherwise, it picks i uniformly at random from $[\ell]$, store (x_i, s_i) in a lookup table, and returns x_i.

The soundness of the lower-bound protocol yields that if $s_i(1-\varepsilon) > |C_1^{-1}(C_1(x_i))|$ then $\mathsf{V}_{\mathsf{SUB}}$ rejects with hight probability. Where the soundness of the upper-bound protocol, yields that if the number of i such that $s_i(1+\varepsilon) < |C_1^{-1}(C_1(x_i))|$ is larger than $1/\varepsilon$, then $\mathsf{V}_{\mathsf{SUB}}$ rejects in at least one of the upper bound protocols with overwhelming probability. It follows that if $\mathsf{V}_{\mathsf{SUB}}$ does not reject with high probability in the interaction, then for a random i it holds that $s_i \approx |C_1^{-1}(C_1(x_i))|$ with probability $1 - \varepsilon$.

Depth 2 queries: On query (C_2, x), $\mathsf{V}_{\mathsf{SUB}}$ checks that (C_1) was asked before and was answered with x, and if not it rejects. Otherwise, it looks up the value of s_x previously stored and uses it to sample a random member of the set $\mathsf{Sib}(x)$ by using the sampling lemma Lemma 3.18. Lemma 3.18 guarantees that this sample is close to uniformly distributed in $C_1^{-1}(C_1(x))$.

Assuming that the prover does not cause the verifier to reject with hight probability, each query of A (or rather each adaptive round of parallel queries) is answered correctly (up to some small statistical deviation), and the proof follows.

□