2-Source Extractors Under Computational Assumptions and Cryptography with Defective Randomness

Yael Tauman Kalai
Microsoft Research
yael@microsoft.com

Xin Li [*]
University of Texas at Austin
lixints@cs.utexas.edu

Anup Rao [†]
Institute for Advanced Study
arao@ias.edu

June 22, 2009

Abstract

We show how to efficiently extract truly random bits from two independent sources of linear min-entropy, under a computational assumption. The assumption we rely on is the existence of an efficiently computable permutation f, such that for any source $X \in \{0,1\}^n$ with linear min-entropy, any circuit of size $\mathrm{poly}(n^{\log n})$ cannot invert $f(X)$ with non-negligible probability.

We use our 2-source extractor to design a computational network extractor protocol. Namely, we design a protocol for a set of processors, each with access to an independent source of linear min-entropy, with the guarantee that at the end of the protocol, each honest processor is left with bits that are computationally indistinguishable from being uniform and private. Our protocol succeeds as long as there are at least two honest players. Our results imply that if such one-way permutations exist, and enhanced trapdoor permutations exist, then secure multiparty computation with imperfect randomness is possible for any number of players, as long as at least two of them are honest.

We also construct a network extractor protocol for the case where each source has only *polynomially-small* min-entropy (n^δ for some constant $\delta > 0$). For this we need at least a constant $u(\delta)$ (which depends on δ) number of honest players, and we need that the one-way permutation is hard to invert even on polynomially small min-entropy sources.

[*] Supported in part by NSF Grant CCF-0634811 and THECB ARP Grant 003658-0113-2007.

[†] Partially supported by NSF grant CCF 0832797. Part of this work was done while this author was visiting Microsoft Research New England.

1 Introduction

Randomness is a useful resource for solving many problems in computer science. In some situations, such as in algorithm design, the use of randomness can lead to simpler and more efficient solutions than can be done deterministically. In other situations, such as in cryptography and distributed computing, randomness can be used to give solutions where deterministic solutions are simply impossible. Thus, it is worthwhile to understand what can be done with and without randomness, and to find the minimal assumptions on the randomness under which these randomized solutions are still viable.

In computer science, it is typically assumed that we have access to perfectly uniform bits. Moreover, in cryptography and distributed computing it is usually assumed that protocol participants have access to randomness that is not only truly uniform, but is also private. In this work, we seek to weaken both types of assumptions.

One generic way to convert schemes that assume perfect private randomness, into schemes which only assume weak private randomness, is to design a *randomness extractor*. This is an algorithm that takes as input a single sample drawn from a *weak source* of randomness, and outputs bits that are close to uniform. A necessary condition on the source is that it must have some entropy: we think of the weak source as a bit-string of length n with some entropy[1] k, and call such a source an (n, k)-source. Unfortunately, it can be shown that for having high entropy alone is not enough for extraction to be feasible — there is no *single* function that can extract a random bit from *every* $(n, n-1)$-source.

A natural model under which randomness extraction is feasible, is to assume that we have access to two or more *independent* sources, each of which has sufficient entropy. Based on past work [CG88, BIW04, BKS$^+$05, Raz05, Bou05, Rao06, BRSW06], we now know how to extract randomness from 2 sources when the entropy in each is at least $.4999n$ [Bou05], from 3 sources if the entropy in each is at least $n^{0.99}$ [Rao06], and from $O(1/\gamma)$ sources if the entropy in each is at least n^{γ} [Rao06, BRSW06]. Although the probabilistic method can be used to show that there is a function that can extract randomness from 2 independent sources even when they have logarithmic entropy, we know of no *efficient* 2-source extractor, even when the entropy is linear. In this work, we make progress towards closing this gap, constructing an efficient 2-source extractor for linear entropy, under a computational assumption.

Although it is impossible to deterministically extract randomness from a single weak source, a sequence of works showed that it is possible to simulate every randomized algorithm with access to a single weak source [VV85, CG88, Zuc96, SSZ98, ACRT99]. However, it is not known how to get an analogous result for protocols in distributed computing or cryptography, where it is essential that each of the processors in the protocol have access to *private* random bits. In fact, the work of Dodis et al. [DOPS04] shows that almost all of the classic cryptographic tasks, including encryption, bit commitment, secret sharing, and secure two-party computation (for nontrivial functions), are actually impossible even with a single $(n, .99n)$-source. This leads us to ask the question: what can be done if each of the processors in the protocol has access to an independent weak source?

It is not immediately clear that extractors for independent sources can be applied, since we need to tolerate adversarial behavior at an unknown subset of the processors, and the extracted bits must remain private even given the information exchanged during the extraction. This question, of

[1] We use a standard measure of entropy called *min-entropy*: a distribution has min-entropy k if all strings have probability at most 2^{-k}.

whether it is possible to do distributed computing with imperfect randomness, was first considered by Goldwasser et al. [GSV05], who showed how to run a Byzantine agreement protocol when each of the processors only has access to a specific kind of independent defective source of randomness (the sources they considered were more restricted than weak sources). In subsequent work, [KLRZ08] showed how to build efficient *network extractor protocols*. These are protocols that have the property that if all the honest processors have access to independent (n, k)-sources, then at the end of the protocol *most* of these honest processors are left with private random bits. Under a non-standard cryptographic assumption, [KLRZ08] designed protocols where *every* honest processor ends up with private randomness, assuming the number of honest processors is larger than $\mathrm{polylog}(n)$, where n is the security parameter. Such a network extractor can be used to do secure multiparty computation, even if each party has access only to a weak source of randomness, as long as the number of honest parties is at least $\mathrm{polylog}(n)$.

The work of [KLRZ08] left open the question of whether secure multiparty computation with imperfect randomness is possible for a constant number of processors (or even for $O(\log n)$ processors). In this work, we give a positive answer to this question, again under a computational assumption.

1.1 Our Results

All of our results in this work are based on the assumption that there exist one-way permutations that are (very) hard to invert, even when the input is sampled from a weak source.

Definition 1.1 (One-Way Functions for Weak Sources)**.** We call a family of polynomial time computable permutations $f : \{0,1\}^n \to \{0,1\}^n$ **one-way for k-sources** if for every (n, k) source X, and every circuit $\mathcal{A}$ of size $2^{O(\log^2 n)}$, $\Pr[\mathcal{A}(f(X)) = X]$ is negligible.

Ideally, we would like to achieve our goals assuming that it is hard to invert these permutations with polynomial sized circuits, but it turns out that our results require the stronger definition above.

Our first result shows that such one-way permutations can be used to obtain 2-source extractors:

Theorem 1.2 (2-Source Extractor)**.** *Suppose that for every $\delta > 0$, there exists a family of one-way permutations for $(n, \delta n)$-sources. Then there is a polynomial time computable 2-source extractor for 2 independent $(n, \delta n)$-sources that extracts $\mathrm{poly}(n)$ bits that are computationally indistinguishable from uniform.*

Note that the first $O(\log n)$ bits of the output of our extractor are actually guaranteed to be statistically indistinguishable from uniform, since every statistical test on such a small number of bits can be efficiently simulated. Thus, even though we make a computational assumption, our conclusion is of an information theoretic nature.

Our next result shows that secure multiparty computation with imperfect randomness is possible for *any* number of processor, as long as there are at least *two* honest processors and each processor has access to an independent source with linear entropy. Secure multiparty computation is also possible if each processor has access to an independent source with polynomially small entropy (n^δ for some constant $\delta > 0$), as long as there are at least $u(\delta)$ honest parties, where here $u = u(\delta)$ is a constant that depends only on δ. We note that this is in contrast with the strong negative results given by Dodis et al. [DOPS04], who show that most of the classic cryptographic tasks, including encryption, bit commitment, secret sharing, and secure two-party computation (for nontrivial functions), are impossible even with a single $(n, .99n)$-source. Thus, in some sense, our results are quite tight.

We obtain these results by constructing computational network extractor protocols where *every* honest player ends up with a private random-looking string. We refer the reader to Section **??** for the formal definition of computational network extractor protocols.

Theorem 1.3 (Network Extractors for Linear Min-Entropy)**.** *Fix a constant $\delta > 0$, and suppose that there exists a family of one-way permutations for $(n, \delta n)$-sources. Then, there is a network extractor protocol where each player takes as input an independent $(n, \delta n)$-source, and as long as there are at least 2 honest players, all the honest players end up with a string that is computationally indistinguishable from being uniform and private.*

Theorem 1.4 (Network Extractors for Polynomial Min-Entropy)**.** *Fix a constant $\delta > 0$, and suppose that there exists a family of one-way permutations for (n, n^{δ})-sources. Then, there exists a constant $u = u(\delta)$ such that there is a network extractor protocol where each player takes as input an independent (n, n^{δ})-source, and as long as there are at least u honest players, all the honest players end up with a string that is computationally indistinguishable from being uniform and private.*

Our network extractor constructions establish that as long as such one-way permutations exist, weak sources are the same as true randomness for the purpose of running cryptographic protocols, as formalized below.

Corollary 1.5. *Fix a constant $\delta > 0$. Assume that there exists a family of one-way permutations for $(n, \delta n)$-sources, and assume that there exists a family of enhanced trapdoor permutations. Then any functionality can be computed securely even if each party has only access to an (independent) $(n, \delta n)$-source, as long as there are at least two honest parties.*

Corollary 1.6. *Fix a constant $\delta > 0$. Assume that there exists a family of one-way permutations for (n, n^{δ})-sources, and assume that there exists a family of enhanced trapdoor permutations. Then there exists a constant $u = u(\delta)$ such that any functionality can be computed securely even if each party has only access to an (independent) (n, n^{δ})-source, as long as there are at least u honest parties.*

1.2 Overview of our Ideas

In this section we shall be slightly inaccurate, in order to easily convey what we consider to be the key ideas in our work. All of our constructions share the same basic structure. They all consist of a sequence of rounds, where in each round i, a string R_i is generated. We will be able to show that there must exist a "good" round j for which R_j is uniform. However, we do not know which of the rounds is the good round. We shall then output is the xor of all the R_i's, and claim that this output is computationally indistinguishable from uniform. We would like to simply say that the r_i's are independent, which would immediately imply that that their xor is indeed uniform. However, we shall not be able to establish this independence. Instead we prove that under the above computational assumption, these r_i's are computationally indistinguishable from being independent, which is enough to get the result we want.

1.2.1 2-Source Extractor

We first note that it is well known how to extract randomness from two independent sources, assuming one of them is a *block source*. A block source X is a source that can be partitioned into two parts $X = (X_1, X_2)$ in such a way that X_1 has entropy δn, and X_2 has entropy δn *even conditioned on any fixing of* $X_1 = x_1$. The entropy in such a source is spread out, and it is well known how to take advantage of such structure. For example, it is known how to extract randomness from a block source $X = (X_1, X_2)$ using an independent weak source Y, as long as the blocks X_1, X_2 and the weak source Y each have entropy δn [BRSW06, RZ08, BKS+05].

Block sources are fairly general, in the sense that *every* weak source can be shown to be a convex combination of block sources — for every source X with linear entropy δn, if X is broken into a sufficiently large ($t = 100/\delta$) number of blocks $X = (X_1, X_2, \ldots, X_t)$, then X is a convex combination of sources, where each element in the convex combination has the structure that there is some index $j \in [t]$ for which (X_j, X) is a block source where each block has linear entropy. Intuitively, each block in the source has at most $\delta n/100$ bits, and so cannot contain all δn bits of entropy.

This fact alone is not enough to apply extractors for block sources, since the index j above is not known ahead of time. Still, we might be tempted to try the following approach:

Naive 2-Source Extractor for (X,Y)

1. Let BExt be an extractor for a block source and an independent weak source
2. Partition $x = (x_1, \ldots, x_t)$.
3. For every i, compute $r_i = \mathsf{BExt}(x_i, x, y)$.
4. Output the bitwise xor $r_1 \oplus \cdots \oplus r_t$.

Since it is no loss of generality to assume that there is some index j for which (X_j, X) is a block source, $R_j = \mathsf{BExt}(X_j, X, Y)$ must be uniform. Unfortunately, the reason this algorithm does not work is that the rest of the candidate random strings R_i are not independent of R_j, and so the output could be a fixed constant even though R_j is uniform.

Our actual construction is a variation of the above construction, where we use computational assumptions to enforce that R_j is independent (in some sense) from the other R_i's. More specifically, we use a one-way permutation for δn-sources to generate independence. This idea was implicit in the work of Goldreich-Levin [**?**] on finding hardcore predicates. There they showed that for any one-way function f, the triplet $(\langle X, R\rangle, R, f(X))$ is computationally indistinguishable from $(U, R, f(X))$, where U is a random bit, and X, R are both uniformly distributed in $\{0,1\}^n$. In other words, they showed that $\langle X, R\rangle$ looks random and *independent* of $(R, f(X))$, even though it may be uniquely determined by $(R, f(X))$. Their construction was an early example of a *reconstructive extractor*, a concept that was subsequently formalized and refined in a sequence of works [NW94, Tre01, TZ04, TUZ01, SU05, Uma05]. We now know of several different constructions of reconstructive extractors. We do not define this concept here, but what is important to know in our application is that every reconstructive extractor RExt must satisfy the property that if f is one-way with respect to a weak source X, then

$$(\mathsf{RExt}(X, R), R, f(X)) \approx (\mathsf{Uniform}, R, f(X)),$$

where $\approx$ denotes computational indistinguishability.

Given such an object, here is how we can use it to build a 2-Source extractor.

Our 2-Source Extractor

1. Let BExt be an extractor for a block source and an independent weak source and RExt be a reconstructive (seeded) extractor. Let f be a one-way permutation for weak sources.
2. Partition $x = (x_1, \ldots, x_t)$.
3. For every i, compute $z_i = \mathsf{BExt}(x_i, x, f^i(y))$.
4. Set $r_i = \mathsf{RExt}(f^i(y), z_i)$.
5. Output the bitwise xor $r_1 \oplus \cdots \oplus r_t$.

Here $f^i(y) = f(f(\cdots f(y) \cdots))$, where f is applied i times. The goal here is to break the dependence (at least in a computational sense) between the R_i's.

The Analysis. To analyze this construction, we need to exploit a strong property of $\mathsf{BExt}(X_1, X_2, Y)$. It turns out that one can show that there is a random variable T on a few bits, such that for every fixing of (T, X_1),

- $(\mathsf{BExt}(X_1, X_2), X)$ is independent of Y.
- $(\mathsf{BExt}(X_1, X_2))$ is uniform.

In particular, since the output of BExt is only a few bits, this means that after fixing (X_1, T), we can fix the output of $\mathsf{BExt}(X_1, X_2)$ and still be left with two independent sources X, Y with high entropy (here we assume the slightly inaccurate fact that fixing a binary string of length l can only reduce the entropy of another variable by l).

Recall that there is some index j for which (X_j, X) is a block source. In the first step of the analysis, we use the properties of BExt described above to fix $Z_1, \ldots, Z_{j-1}$ and $R_1, \ldots, R_{j-1}$. We claim that even after this fixing, (X_j, X) is a block source that is independent of the source Y with linear entropy. We do this by fixing each of the $Z_1, \ldots, Z_{j-1}$'s one by one. Each such fixing maintains the independence we want, yet does not reduce the entropy of the sources by much, since the Z_i's are short. Once all the Z_i's are fixed, the corresponding R_i's are deterministic functions of Y that output only a few bits, so we can fix them without reducing the entropy in Y by much. Care must be taken that all of these fixings do not ruin the entropy in X_j (in particular, fixing $X_1, \ldots, X_{j-1}$ should not ruin the entropy in X_j), but it turns out that this can be done.

We get that after all these fixings, Z_j must be uniform and independent of Y. Thus, by the properties of reconstructive extractors, the following two distributions are computationally indistinguishable:

$$(R_j, Z_j, f^{j+1}(Y)) \approx (\mathsf{Uniform}, Z_j, f^{j+1}(Y)).$$

In fact, we can actually prove the stronger statement that

$$(R_j, X, f^{j+1}(Y)) \approx (\mathsf{Uniform}, X, f^{j+1}(Y)).$$

Observe that information theoretically this is very far from true. In fact, R_j is a deterministic function of $(X, f^{j+1}(Y))$. Finally, since $(R_{j+1}, \dots, R_t)$ are all efficiently computable from $(X, f^{j+1}(Y))$, we obtain

$$(X, f^{j+1}(Y), R_j) \approx (X, f^{j+1}(Y), \mathsf{Uniform}),$$

which implies that the output of our extractor is computationally indistinguishable from uniform.

In fact, our proof shows that the extractor is *strong* — the output looks uniform even if one of the inputs is known.

1.2.2 Network Extractor Protocols

We construct two network extractor protocols. One for the case where each player has an independent source with linear entropy δn, and one for the case where each player has an independent source with polynomially-small entropy n^{δ}. We start with the former.

We first present a protocol where all the honest players *except one* end up with private randomness. Note that if we knew of one player j that is honest, then the protocol would be very simple: Player j will simply reveal his source, and all other players would apply the 2-source extractor TExt (presented above) to this source and to their own source. The fact TExt is a strong extractor immediately implies that all the honest players, except for player j, would end up with private randomness. However, since we don't know of any player that is honest, it is tempting to try the following approach.

Naive Network Extractor Protocol for Linear Min-Entropy

1. The protocol proceeds in p rounds, where p is the number of players. In round i:
   (a) Player i sends x_i to all other players.
   (b) Each player ℓ computes $r_\ell^i = \mathsf{TExt}(x_i, x_\ell)$.
2. Each player i outputs the bitwise xor $r_i^1 \oplus \cdots \oplus r_i^p$.

Let j denote the first honest player. Then, for every player i different than j, r_i^j is uniform. Despite this, the output of player i may not be random, and may even be a fixed constant. As before, the problem is that the random variables $r_i^1, \dots, r_i^p$ are not independent, and a malicious adversary can actually cause the output to be a fixed constant.

As in our 2-source extractor construction, the idea is to get around this problem by using computational assumptions. To this end, each player $i\ell$, instead of using the same source X_ℓ in each round, will use the source $f^i(X_\ell)$ in round ℓ. However, for this approach to work we need the guarantee that

$$(\mathsf{TExt}(X_i, X_\ell), X_\ell, f(X_i)) \approx (\mathsf{Uniform}, X_\ell, f(X_i)).$$

Our extractor TExt does not satisfy this, but luckily our extractor does satisfty the following (similar) guarantee:

$$(\mathsf{TExt}(X_i, X_j), X_j, f^{t+1}(X_i)) \approx (\mathsf{Uniform}, X_j, f^{t+1}(X_i)),$$

where t is a constant that depends on δ.

So, instead we consider the following network extractor protocol, which has the guarantee that all the honest players, except for the first one, end up with private random-looking strings.

Lossy Network Extractor Protocol for Linear Min-Entropy

1. Let $g = f^{t+1}$. The protocol proceeds in p rounds, where p is the number of players. In round i:

   (a) Player i sends $g^i(x_i)$ to all other players.

   (b) Each player ℓ computes $r^i_\ell = \mathsf{TExt}(g^i(x_i), g^i(x_\ell))$.

2. Each player i outputs the bitwise xor $r^1_i \oplus \cdots \oplus r^p_i$.

Now we can prove that all the honest players, except player j (who is the first honest player), end up with private random-looking strings. The analysis proceeds in three steps.

1. We first fix all sources sent before round j, and we fix $\{r^i_\ell\}_{\ell\in[p],i<j}$, which were all computed before round j. We claim that even conditioned on all these fixings, the sources are still independent, and with high probability they all have "enough" entropy left.

2. Next, we claim that the strings $\{r^j_\ell\}$ of all the honest players $\ell \neq j$, are independent and uniformly distributed.

3. Finally, we claim that the rest of the r^i_ℓ for $i > j$ are (computationally) independent of $\{r^j_\ell\}$, which implies that the output of all the honest players, except player j, are computationally indistinguishable from random. For this we use the fact that for any two independent variables Y_i and Y_j with "sufficient" entropy

$$(\mathsf{TExt}(Y_i, Y_j), Y_j, g(Y_i)) \approx (\mathsf{Uniform}, Y_j, g(Y_i)). \tag{1}$$

Note that in the protocol above, player j, who is the first honest player, does not necessarily end with private randomness. To fix this, we add another phase to the protocol. So, the protocol consists of two phases. In the first phase, the players run the (lossy) protocol presented above. In the second phase, the idea is that all the honest players use their (supposedly) random string, generated in the first phase, to run a coin flipping protocol and generate a public random-looking seed V. Recall that we assumed that there are at least two honest players, therefore there is at least one honest player besides player j. Thus, V is indeed random-looking. Finally, each player i will extract randomness from his own source X_i using the seed V.

This approach would indeed work if there were at least *three* honest players, since in that case we could argue that V is random-looking and is *independent* of each of the sources X_i. Thus, we could use it to extract (private) randomness from each source.

However, if there are only *two* honest players then this approach does not seem to work, since in this case we cannot argue that V is independent of all the sources. Indeed if there is a single honest player ℓ besides player j (who is the first honest player) then it may be the case that V depends on the source of player ℓ. This is the case since player ℓ maybe the only player who used "good" randomness for the coin-flipping protocol. As before, we get around this dependence by using reconstructive properties of extractors.

Our Final Network Extractor Protocol for Linear Min-Entropy.

1. **Phase 1.** This phase proceeds in p rounds, where p is the number of players. In round i:
   (a) Player i sends $g^{2i}(x_i)$ to all other players.
   (b) Each player ℓ computes $r_\ell^i = \mathsf{TExt}(g^{2i}(x_i), g^{2i}(x_\ell))$.

   At the end of this phase, each player i outputs $r_i = r_i^1 \oplus \cdots \oplus r_i^p$.

2. **Phase 2.** Each player i uses its (supposedly) random string r_i, generated in Phase 1, to run a secure coin flipping protocol. Finally, each player i outputs $Z_i = \mathsf{RExt}(g^{(2i-1)}(X_i), V)$.

We claim that even if there are only two honest players ℓ and j, player ℓ (and player j) still end up with private randomness. The reason why player j ends up with private randomness is quite straightforward: It follows from the fact that V is random and is independent of X_j. The reason why player ℓ ends up with private randomness is more involved. On a very high-level, the proof follows the following two steps.

1. First, we use the reconstructive property of our 2-source extractor TExt (Equation 1), to argue that R_ℓ looks random, even conditioned on all the sources of all the other players *and* conditioned on $g^{2j+1}(X_\ell)$. We stress that the above statement is a computational statement, and cannot hold information theoretically, since R_ℓ is uniquely determined by $g^{2j+1}(X_\ell)$ and all the other sources.

2. Then, we use the reconstructive property of RExt, together with the conclusion of step 1 above, to argue that indeed player ℓ ends up with private randomness.

We refer the reader to Section **??** for details of the proof.

Our Network Extractor Protocol for Polynomially-Small Min-Entropy. Finally, we construct a network extractor for the case where each player has an independent (n, n^δ)-source (for some constant $\delta > 0$). This protocol is very similar (in spirit) to the protocol above, though is significantly more complicated. The reason for the complication is that in this setting we cannot use a 2-source extractor, since we do not know of such an extractor for the case where the entropy is polynomially small. Thus, instead, we use a multi-source extractor IExt that extracts randomness from $u = u(\delta)$ independent (n, n^δ)-sources [**?**, **?**].

As before, the protocol proceeds in two phases. In the first phase, all the honest players except u of them, end up with private randomness. In the second phase, all the players run a secure coin-tossing protocol using the (supposedly) random string they obtained in the first phase, to get a public random seed V. Then, each player i will use a reconstructive extractor to extract randomness from his source (more precisely, from a function of his source), using the random seed V. To argue that all honest players end up with private randomness we need to assume that there are at least $u + 1$ honest players. We refer the reader to Section D for details.

Roadmap. Due to lack of space we only give a formal description of our 2-source extractor in the body of the paper. The preliminaries section is deferred to Appendix A. The formal descriptions and proofs of the network extractor protocols for the linear entropy setting and the polynomial entropy setting are deferred to Appendix C and Appendix **??**, respectively.

2 Two-Source Extractor

In this section we present our construction of a two-source extractor under computational assumptions. We assume one of the sources has linear min-entropy, while the other one can have smaller min-entropy.

2.1 Ingredients

Our construction uses for ingredients: Zuc, $\mathsf{BasicExt}$, Raz, RExt, all of which are some form of extractors. In order to state the properties of these extractors, we need the following definition.

Definition 2.1. A source $X = (X_1, \cdots, X_t)$ is $t \times r$ **somewhere-random** (SR-source, for short) if each X_i takes values in $\{0,1\}^r$ and there is an i such that X_i is uniformly distributed. It is $(t \times r)$ **k-somewhere-random** (k-SR-source for short) if each X_i takes values in $\{0,1\}^r$ and there is an i such that X_i has min-entropy k.

We next state state the four theorems that we rely on.

Theorem 2.2 ([?]). *For every constant $0 < \alpha < 1$ there exists a function*

$$\mathrm{Zuc} : \{0,1\}^n \to \{0,1\}^{c \times \ell}$$

where $c = \mathrm{poly}(1/\alpha)$ is a constant and $\ell = \frac{n}{\mathrm{poly}(1/\alpha)}$, such that for every $(n, \alpha n)$-source X, $\mathrm{Zuc}(X)$ is $2^{-\Omega(n)}$-close to a $(c \times \ell)0.9\ell$-somewhere random source.

Theorem 2.3 ([BRSW06]). *There exist constants $\alpha, \beta < 1$ such that for every $n, k(n)$ with $k > \log^{10} n$, and constant $0 < \gamma < 1/2$, there is a polynomial time computable function $\mathsf{BasicExt} : \{0,1\}^n \times \{0,1\}^{k^{\gamma+1}} \to \{0,1\}^m$ s.t. if X is an (n,k) source and Y is a $(k^\gamma \times k)$ $(k - k^\beta)$-SR-source,*

$$|(Y, \mathsf{BasicExt}(X,Y)) - (Y, U_m)| < \epsilon$$

and

$$|(X, \mathsf{BasicExt}(X,Y)) - (X, U_m)| < \epsilon$$

where U_m is independent of X, Y, $m = k - k^{\Omega(1)}$ and $\epsilon = 2^{-k^\alpha}$.

Theorem 2.4 ([Raz05]). *For any n_1, n_2, k_1, k_2, m and any $0 < \delta < 1/2$ with*

- $n_1 \geq 6 \log n_1 + 2 \log n_2$
- $k_1 \geq (0.5 + \delta) n_1 + 3 \log n_1 + \log n_2$
- $k_2 \geq 5 \log(n_1 - k_1)$
- $m \leq \delta \min[n_1/8, k_2/40] - 1$

There is a polynomial time computable strong 2-source extractor

$$\mathsf{Raz} : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \to \{0,1\}^m$$

for min-entropy k_1, k_2 with error $2^{-1.5m}$.

Theorem 2.5. *For every n, k, ϵ with $k = n^{\Omega(1)}$, there is a polynomial time computable function $\mathsf{RExt} : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ such that $t = O(\log(n/\epsilon))$, $m = k^{\Omega(1)}$ and if f is a one way function for $k/3$ sources and X is an (n,k) source,*

$$(f(X), \mathsf{RExt}(X,U), U) \approx (f(X), \mathsf{Uniform})$$

2.2 Construction

Let X and Y be two independent weak sources, where X is an $(n, \alpha n)$-source for some constant $\alpha > 0$ (i.e., X has linear min-entropy); and Y is an (n, k) source, where $k \geq \mathrm{polylog}(n)$.

We first define an extractor for a block source and an independent general source, as in [BRSW06]:

$$\mathsf{BExt} : \{0,1\}^{n_1} \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$$

- **Ingredients**:
  - $\mathrm{Zuc} : \{0,1\}^{\alpha n/2} \to \{0,1\}^{c \times \ell}$ a function as in Theorem A.16, where $c = \mathrm{poly}(1/\alpha)$ and $\ell = \frac{n}{\mathrm{poly}(1/\alpha)}$.
  - $\mathsf{Raz} : \{0,1\}^\ell \times \{0,1\}^n \to \{0,1\}^s$ a strong 2-source extractor as in Theorem A.23, where one source is a $(\ell, 0.9\ell)$-source and the other is an $(n, 0.9k)$-source (where k is the min-entropy of Y).
  - $\mathsf{BasicExt} : \{0,1\}^n \times \{0,1\}^{s^\gamma \times s} \to \{0,1\}^d$ the extractor as in Theorem A.22, where $s = \mathrm{polylog}(n)$.

- $\mathsf{BExt}(x_1, x, y)$:

  1. Compute $v = \mathrm{Zuc}(x_1)$.
  2. Apply the strong 2-source extractor Raz to each row of v and y. Denote the resulting matrix by sr. Note each row of sr is of length $s = \mathrm{polylog}(n)$.
  3. Output $\mathsf{BasicExt}(x, sr)$

Next we use this block source extractor to extract

$$\mathsf{TExt} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$$

- **Ingredients**:
  - $\mathsf{RExt} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ a $(0.9k, \epsilon)$-extractor as in Theorem A.19, where $\epsilon = \frac{1}{\mathrm{poly}(n^{\log n})}$ and $d = O(\log(n/\epsilon)) = \mathrm{polylog}(n)$.
  - f, a $(s, 0.9k, \epsilon)$ one-way permutation

- $\mathsf{TExt}(x, y)$:

  Partition x, into $t = \frac{4}{\alpha}$ equally sized parts[2] $x = (x_1, \ldots, x_t)$.

  For every $i = 1, \ldots, t$,

  1. Compute $r_i = \mathsf{BExt}(x_i, x, f^{(i)}(y))$, and assume without loss of generality that r_i is a d length string where $d = \mathrm{polylog}(n)$ is the seed length for the extractor RExt.
  2. Compute $z_i = \mathsf{RExt}(f^{(i)}(y), r_i)$, and assume without loss of generality that z_i is of length $m \leq \mathrm{polylog}(n)$.

[2] For the sake of simplicity, we assume t is an integer.

Output the bitwise parity $z = \oplus_{i=1}^{t} z_i$.

Theorem 2.6. *Fix a constant* $\alpha > 0$ *and parameters* $t = \frac{4}{\alpha}$ *and* $k \geq n^{\Omega(1)}$*. Assume that there exists a permutation* $f : \{0,1\}^n \to \{0,1\}^n$ *such that for any* $(n, 0.9k)$*-source* Y*, any non-uniform adversary that runs in time* $\mathrm{poly}(n^{\log n}, 2^m)$ *can invert* $f(Y)$ *with only negligible probability. Then* $\mathsf{TExt} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$ *described above is a computational 2-source extractor such that for any* $(n, \alpha n)$*-source* X*, and any* (n,k)*-source* Y *that is independent of* X*,*

$$(\mathsf{TExt}(X,Y), X) \approx (U_m, X)$$

The proof of this Theorem is deferred to Appendix **??**.

3 Acknowledgements

We thank Chris Umans and David Zuckerman for useful discussions.

References

[ACRT99] Alexander E. Andreev, Andrea E. F. Clementi, José D. P. Rolim, and Luca Trevisan. Weak random sources, hitting sets, and BPP simulations. *SIAM Journal on Computing*, 28(6):2103–2116, 1999.

[BIW04] Boaz Barak, R. Impagliazzo, and Avi Wigderson. Extracting randomness using few independent sources. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 384–393, 2004.

[BKS+05] Boaz Barak, Guy Kindler, Ronen Shaltiel, Benny Sudakov, and Avi Wigderson. Simulating independence: New constructions of condensers, Ramsey graphs, dispersers, and extractors. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 1–10, 2005.

[BRSW06] Boaz Barak, Anup Rao, Ronen Shaltiel, and Avi Wigderson. 2 source dispersers for $n^{o(1)}$ entropy and Ramsey graphs beating the Frankl-Wilson construction. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006.

[Bou05] Jean Bourgain. More on the sum-product phenomenon in prime fields and its applications. *International Journal of Number Theory*, 1:1–32, 2005.

[CG88] Benny Chor and Oded Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. *SIAM Journal on Computing*, 17(2):230–261, 1988.

[DOPS04] Yevgeniy Dodis, Shien Jin Ong, Manoj Prabhakaran, and Amit Sahai. On the (im)possibility of cryptography with imperfect randomness. In *FOCS*, pages 196–205. IEEE Computer Society, 2004.

[GSV05] Shafi Goldwasser, Madhu Sudan, and Vinod Vaikuntanathan. Distributed computing with imperfect randomness. In *Distributed Computing, 19th International Conference, DISC 2005, Cracow, Poland, September 26-29, 2005, Proceedings*, volume 3724 of *Lecture Notes in Computer Science*, pages 288–302. Springer, 2005.

[KLRZ08] Yael Tauman Kalai, Xin Li, Anup Rao, and David Zuckerman. Network extractor protocols. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*. IEEE Computer Society, 2008.

[MW97] Ueli Maurer and Stefan Wolf. Privacy amplification secure against active adversaries. In Burton S. Kaliski Jr., editor, *Advances in Cryptology — CRYPTO '97*, volume 1294 of *Lecture Notes in Computer Science*, pages 307–321. Springer-Verlag, August 1997.

[NW94] Noam Nisan and Avi Wigderson. Hardness vs randomness. *Journal of Computer and System Sciences*, 49(2):149–167, October 1994.

[Rao06] Anup Rao. Extractors for a constant number of polynomially small min-entropy independent sources. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006.

[RZ08] Anup Rao and David Zuckerman. Extractors for 3 uneven length sources. In *RANDOM 2008, 12th International Workshop on Randomization and Approximation Techniques in Computer Science*, 2008.

[Raz05] Ran Raz. Extractors with weak random seeds. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 11–20, 2005.

[SSZ98] Michael Saks, Aravind Srinivasan, and Shiyu Zhou. Explicit OR-dispersers with polylog degree. *Journal of the ACM*, 45:123–154, 1998.

[SU05] Ronen Shaltiel and Chris Umans. Simple extractors for all min-entropies and a new pseudorandom generator. *Journal of the ACM*, 52:172–216, 2005.

[TUZ01] Amnon Ta-Shma, Chris Umans, and David Zuckerman. Loss-less condensers, unbalanced expanders, and extractors. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pages 143–152, 2001.

[TZ04] Amnon Ta-Shma and David Zuckerman. Extractor codes. *IEEE Transactions on Information Theory*, 50, 2004.

[Tre01] Luca Trevisan. Extractors and pseudorandom generators. *Journal of the ACM*, pages 860–879, 2001.

[Uma05] Christopher Umans. Reconstructive dispersers and hitting set generators. In Chandra Chekuri, Klaus Jansen, José D. P. Rolim, and Luca Trevisan, editors, *APPROX-RANDOM*, volume 3624 of *Lecture Notes in Computer Science*, pages 460–471. Springer, 2005.

[VV85] Umesh V. Vazirani and Vijay V. Vazirani. Random polynomial time is equal to slightly-random polynomial time. In *Proceedings of the 26th Annual IEEE Symposium on Foundations of Computer Science*, pages 417–428, 1985.

[Zuc96] David Zuckerman. Simulating BPP using a general weak random source. *Algorithmica*, 16:367–391, 1996.

A Preliminaries

A.1 Basic Definitions

Definition A.1. The *min-entropy* of a random variable X is defined as

$$H_\infty(X) = min_{x \in \mathsf{supp}(X)}\{-\log_2 \Pr[X = x]\}.$$

We say X is an (n, k)-source if X is a random variable on $\{0,1\}^n$ and $H_\infty(X) \geq k$.

Definition A.2. A function $\mu(\cdot)$ is **negligible** if for every polynomial $q(\cdot)$ there exists a value N such that for all $n > N$ it holds that $\mu(n) < 1/q(n)$.

Definition A.3. Let $\mathcal{D} = \{D_n\}_{n\in\mathbb{N}}$ and $\mathcal{F} = \{F_n\}_{N\in\mathbb{N}}$ be two distribution ensembles. We say that $\mathcal{D}$ and $\mathcal{F}$ are **computationally indistinguishable**, denoted by $\mathcal{D} \approx \mathcal{F}$, if for every non-uniform algorithm $\mathcal{A}$ running in time poly(n) there exists a negligible function ϵ such that for every $n \in \mathbb{N}$,

$$|\Pr[\mathcal{A}(D_n) = 1] - \Pr[\mathcal{A}(F_n) = 1]| \leq \epsilon(n).$$

Remark. Often we abuse notation, and let $D_n \approx F_n$ denote the fact that the two ensembles are computationally indistinguishable.

Definition A.4. Let D and F be two distributions on a set S. Their **statistical distance** is

$$|D - F| \stackrel{def}{=} \max_{T\subseteq S}(|D(T) - F(T)|) = \frac{1}{2}\sum_{s\in S}|D(s) - F(s)|$$

If $|D - F| \leq \epsilon$ we say that D is ϵ-*close* to F.

Definition A.5. If $\mathcal{D} = \{D_n\}_{n\in\mathbb{N}}$ and $\mathcal{F} = \{F_n\}_{N\in\mathbb{N}}$ are two distribution ensembles, and there exists a negligible function $\epsilon(n)$ such that for every $n \in \mathbb{N}$,

$$|D_n - F_n| \leq \epsilon(n),$$

then we say that $\mathcal{D}$ and $\mathcal{F}$ are **statistically close**, and denote it by

$$\mathcal{D} \equiv \mathcal{F}.$$

Remark. Often we abuse notation, and let $D_n \equiv F_n$ denote the fact that the two ensembles are statistically close.

This measure of distance is nice because it is robust in the sense that if two distributions are close in this distance, then applying any functions to them cannot make them go further apart.

Proposition A.6. *Let D and F be any two distributions over a set S s.t. $|D - F| \leq \epsilon$. Let g be any function on S. Then $|g(D) - g(F)| \leq \epsilon$.*

Lemma A.7. *Let $\{X_n\}$ and $\{Y_n\}$ be two distribution ensembles. Let $E = \{E_n\}$ be a sequence of events for which there exists a negligible function ϵ such that $\Pr[E_n] = 1 - \epsilon(n)$. Then $\{X_n|E_n\} \approx \{Y_n|E_n\}$ implies that $\{X_n\} \approx \{Y_n\}$.*

We will also use the the following generalization of Lemma A.7.

Lemma A.8. *Let $\{X_n\}$ and $\{Y_n\}$ be two distribution ensembles. Let J be a set such that for every $j \in J$, $E^j = \{E^j_n\}$ is a sequence of events for which $\{X_n|E^j_n\} \approx \{Y_n|E^j_n\}$. Then, if there exists a negligible function ϵ such that $\Pr[\cup_{j\in J} E^j_n] = 1 - \epsilon(n)$, then $\{X_n\} \approx \{Y_n\}$.*

A.2 Block Sources and Conditional entropy.

A block source is a source broken up into a sequence of blocks, with the property that each block has min-entropy even conditioned on previous blocks.

Definition A.9 (Conditional Min-Entropy)**.** Given random variables A, B in the same probability space, we define the conditional min-entropy

$$H_\infty(A|B) = \min_b H_\infty(A|B = b)$$

Definition A.10 (Block sources)**.** A distribution $X = X_1, X_2, \cdots, X_{\mathsf{C}}$ is called a $(k_1, k_2, \ldots, k_{\mathsf{C}})$-block source if for all $i = 1, \ldots, \mathsf{C}$, we have that $H_\infty(X_i|X_{i-1}, \ldots, X_1) \geq k_i$. If $k_i = k$ for every i, we say that X is a k-block source.

Let $X = X_1, \cdots, X_{\mathsf{t}}$ be a random variable over $\{0,1\}^n$ divided into t blocks in some way, and $x_1, \ldots, x_i$ are some strings with $0 \leq i < \mathsf{t}$. We use the notation $X|x_1, \ldots, x_i$ to denote the random variable X conditioned on $X_1 = x_1$,…,$X_i = x_i$. For $1 \leq i < j \leq \mathsf{t}$, we denote by $X_{i,\ldots,j}$ the projection of X onto the blocks $X_i, \ldots, X_j$.

Next we show that any weak source with linear min-entropy can be divided into a constant number of blocks, such that the source is close to a convex combination of block sources.

Lemma A.11. *Let X be an $(n, \alpha n)$ source for some constant $0 < \alpha < 1$. Let $t = \frac{4}{\alpha}$. Divide X evenly into t blocks $X = X_1 \circ X_2 \circ \cdots \circ X_t$. Then X is $2^{-\Omega(n)}$-close to being a convex combination of sources $\{X^j\}_{j\in J}$ such that for every j there exists $g \in [t]$ for which*

- *$X^j_1, \ldots, X^j_{g-1}$ is fixed.*
- *$H_\infty(X^j_g) \geq \frac{\alpha^2}{6}$.*
- *$H_\infty(X|X^j_g) \geq \frac{\alpha^2}{6}$.*

The proof of this theorem is deferred to Appendix E.

We use the following standard lemma about conditional min-entropy. (For a proof, we refer the reader to the proof of Lemma 5 in [MW97]).

Lemma A.12. *Let X and Y be random variables and let $\mathcal{Y}$ denote the range of Y. Then for all $\epsilon > 0$*

$$\Pr_Y\left[H_\infty(X|Y = y) \geq H_\infty(X) - \log|\mathcal{Y}| - \log\left(\frac{1}{\epsilon}\right)\right] \geq 1 - \epsilon$$

Sometimes a random variable may only be close to having a certain amount of min-entropy, and we have the following lemma, which can be viewed as a generalization of Lemma A.12.

Lemma A.13. *Let X be an (n,k)-source and X' be a random variable such that $|X - X'| < \epsilon$. Let Z be another random variable and $\mathcal{Z}$ denote the range of Z. Then for all $\epsilon_1 > 0$*

$$\Pr_{Z}\left[(X'|Z=z) \text{ is } \frac{\epsilon|\mathcal{Z}|}{\epsilon_1} \text{ close to having min-entropy } k - \log|\mathcal{Z}| - \log\left(\frac{1}{\epsilon_1}\right)\right] \geq 1 - \epsilon_1$$

Proof. For a particular $Z = z$, $\Pr[X' = x'|Z = z] = \frac{\Pr[X'=x', Z=z]}{\Pr[Z=z]}$. Since X is an (n,k)-source X must have size of support at least 2^k. Choose a subset S in the support of size $\epsilon_1 2^k/|\mathcal{Z}|$ and let $\bar{X}$ be the source that is uniformly distributed on S. Then $H_\infty(\bar{X}) = k - \log|\mathcal{Z}| - \log(1/\epsilon_1)$. Let R be the set $\{r \in S : \Pr[X' = r|Z = z] > |\mathcal{Z}|/(\epsilon_1 2^k)\}$, then

$$|(X'|Z=z) - \bar{X}| = \sum_{r \in R}(\Pr[X' = r|Z = z] - |\mathcal{Z}|/(\epsilon_1 2^k)).$$

If $\Pr[Z = z] > \frac{\epsilon_1}{|\mathcal{Z}|}$, then

$$|X'|(Z=z) - \bar{X}| < \sum_{r \in R} \frac{\Pr[X' = r, Z = z] - 2^{-k}}{\epsilon_1/|\mathcal{Z}|} \leq \frac{\sum_{r\in R}\Pr[X' = r] - \Pr[X = r]}{\epsilon_1/|\mathcal{Z}|} \leq \frac{\epsilon}{\epsilon_1/|\mathcal{Z}|} = \frac{\epsilon|\mathcal{Z}|}{\epsilon_1}.$$

The probability this does not happen is at most $\frac{\epsilon_1}{|\mathcal{Z}|}|\mathcal{Z}| = \epsilon_1$. ■

A.3 Somewhere Random Sources

Definition A.14. A source $X = (X_1, \cdots, X_t)$ is $t \times r$ **somewhere-random** (SR-source, for short) if each X_i takes values in $\{0,1\}^r$ and there is an i such that X_i is uniformly distributed.

Note that every $t \times r$ somewhere random source must have min-entropy at least r, since the random row itself has min-entropy r.

Definition A.15. A source $X = (X_1, \cdots, X_t)$ is $(t \times r)$ **k-somewhere-random** (k-SR-source for short) if each X_i takes values in $\{0,1\}^r$ and there is an i such that X_i has min-entropy k.

Theorem A.16 ([?])**.** *For every constant $0 < \alpha < 1$ there exists a function*

$$\mathrm{Zuc} : \{0,1\}^n \to \{0,1\}^{c \times \ell}$$

where $c = \mathrm{poly}(1/\alpha)$ is a constant and $\ell = \frac{n}{\mathrm{poly}(1/\alpha)}$, such that for every $(n, \alpha n)$-source X, $\mathrm{Zuc}(X)$ is $2^{-\Omega(n)}$-close to a $(c \times \ell)0.9\ell$-somewhere random source.

A.4 Seeded Extractors and Independent Source Extractors

Definition A.17. A function $\mathsf{IExt} : (\{0,1\}^n)^u \to \{0,1\}^m$ is a (k,ϵ) *extractor for* C *independent sources* if for any independent (n,k) sources $X^1, \ldots, X^{\mathsf{C}}$ we have

$$|\mathsf{IExt}(X^1, \ldots, X^{\mathsf{C}}) - U_m| < \epsilon$$

A *seeded extractor* is just a special case of the above definition:

Definition A.18 (Seeded Extractor)**.** A function $\mathsf{Ext} : \{0,1\}^{n_1 \times n_2} \to \{0,1\}^m$ is called a (k,ϵ)-*seeded extractor* if it is a 2-source extractor with k, n_2 min-entropy requirement and error ϵ.

A.5 Reconstructive Extractors and One Way Functions

The following theorem follows easily from known extractor constructions. We defer the arguments to Appendix F.

Theorem A.19. *For every $n, k, \epsilon, \epsilon_1$ with $k = n^{\Omega(1)}$ and $\epsilon_1 \geq \epsilon \geq 2^{-\sqrt{k}}$, there is a polynomial time computable function* $\mathsf{RExt} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *with $d = O(\log(n/\epsilon))$, $m = k^{\Omega(1)}$ such that the following holds: For any (n,k)-source X and any deterministic function f on $\{0,1\}^n$, if there exists a non-uniform adversary $\mathcal{A}$ that distinguishes* $(\mathsf{RExt}(X, U_d), U_d, f(X))$ *and* $(U_m, U_d, f(X))$ *with probability $2\epsilon_1$, then there exists another non-uniform adversary $\mathcal{B}$ of size* $\mathrm{poly}(n, 1/\epsilon) \cdot Size(\mathcal{A})$ *and an $(n, k/3)$-source $\bar{X}$ such that $\mathcal{B}$ inverts $f(\bar{X})$ with probability at least $\epsilon_1/4$.*

Lemma A.20. *Let X, Y be two independent random variables on $\{0,1\}^n$ and Z be a random variable on $\{0,1\}^m$ that is independent of (X, Y). Let $f : \{0,1\}^n \to \{0,1\}^d$ and $g : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ be two deterministic functions. Let $R = f(X)$. If there exists a non-uniform adversary $\mathcal{A}$ that distinguishes between $(g(Y,R), R, X, Y)$ and (Z, R, X, Y) with probability ϵ, then there exists another non-uniform adversary $\mathcal{B}$ of size $2^d \cdot n \cdot Size(\mathcal{A})$ that distinguishes between $(g(Y,R), R, Y)$ and (Z, R, Y) with probability at least ϵ.*

Proof. Assume without loss of generality that

$$\Pr[\mathcal{A}(g(Y,R), R, X, Y) = 1] - \Pr[\mathcal{A}(Z, R, X, Y) = 1] \geq \epsilon.$$

Note that R is a deterministic function of X, thus for any fixing of $R = r$, $(g(Y,R), Z, Y)|(R = r)$ is independent of $X|(R = r)$. Therefore, for every fixing of $R = r$, there exists a fixing of $X|(R = r)$ and a non-uniform adversary $\mathcal{A}_r$, that has this fixing hardwired into it and emulates $\mathcal{A}$ w.r.t. this fixing, s.t.

$$\begin{aligned}&\Pr[\mathcal{A}_r(g(Y,R), Y) = 1 | R = r] - \Pr[\mathcal{A}_r(Z, Y) = 1 | R = r] \geq \\ &\Pr[\mathcal{A}(g(Y,R), R, X, Y) = 1 | R = r] - \Pr[\mathcal{A}(Z, R, X, Y) = 1 | R = r].\end{aligned}$$

Let $\mathcal{B}$ be an adversary that on input $(g(Y,r), r, Y)$ emulates $\mathcal{A}_r(g(Y,r), Y)$. Then we have

$$\begin{aligned}&\Pr[\mathcal{B}(g(Y,R), R, Y) = 1] - \Pr[\mathcal{B}(Z, R, Y) = 1] = \\ &\sum_r \Pr[R = r] \left(\Pr[\mathcal{A}_r(g(Y,R), Y) = 1 | R = r] - \Pr[\mathcal{A}_r(Z, Y) = 1 | R = r]\right) \geq \\ &\sum_r \Pr[R = r] \left(\Pr[\mathcal{A}(g(Y,R), R, X, Y) = 1 | R = r] - \Pr[\mathcal{A}(Z, R, X, Y) = 1 | R = r]\right) = \\ &\Pr[\mathcal{A}(g(Y,R), R, X, Y) = 1] - \Pr[\mathcal{A}(Z, R, X, Y) = 1] \geq \epsilon.\end{aligned}$$

Moreover, $\mathcal{B}$ is of size $2^d \cdot n \cdot Size(\mathcal{A})$. ■

A.6 Previous Work that We Use

Theorem A.21 ([Rao06, BRSW06])**.** *There exist constants $c > 0$ and c' such that for every n, k with $k = k(n) = \Omega(\log^4 n)$ there exists a polynomial time computable function* $\mathsf{IExt} : (\{0,1\}^n)^u \to \{0,1\}^k$ *with $u \leq c' \frac{\log n}{\log k}$ s.t. if $X^1, X^2, \ldots, X^u$ are independent (n,k) sources then*

$$|\mathsf{IExt}(X^1, \ldots, X^u) - U_k| < 2^{-k^c}.$$

Moreover, IExt *is a strong extractor.*

Theorem A.22 ([BRSW06]). *There exist constants $\alpha, \beta < 1$ such that for every $n, k(n)$ with $k > \log^{10} n$, and constant $0 < \gamma < 1/2$, there is a polynomial time computable function* $\mathsf{BasicExt} : \{0,1\}^n \times \{0,1\}^{k^{\gamma+1}} \to \{0,1\}^m$ *s.t. if X is an (n,k) source and Y is a $(k^\gamma \times k)$ $(k - k^\beta)$-SR-source,*

$$|(Y, \mathsf{BasicExt}(X,Y)) - (Y, U_m)| < \epsilon$$

and

$$|(X, \mathsf{BasicExt}(X,Y)) - (X, U_m)| < \epsilon$$

where U_m is independent of X, Y, $m = k - k^{\Omega(1)}$ and $\epsilon = 2^{-k^\alpha}$.

Theorem A.23 ([Raz05]). *For any n_1, n_2, k_1, k_2, m and any $0 < \delta < 1/2$ with*

- $n_1 \geq 6 \log n_1 + 2 \log n_2$
- $k_1 \geq (0.5 + \delta) n_1 + 3 \log n_1 + \log n_2$
- $k_2 \geq 5 \log(n_1 - k_1)$
- $m \leq \delta \min[n_1/8, k_2/40] - 1$

There is a polynomial time computable strong 2-source extractor

$$\mathsf{Raz} : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \to \{0,1\}^m$$

for min-entropy k_1, k_2 with error $2^{-1.5m}$.

B Computational Two-Source Extractor

In this section we present our construction of a computational two-source extractor. We assume one of the sources has linear min-entropy, while the other one can have smaller min-entropy.

B.1 Construction

Let X and Y be two independent weak sources, where X is an $(n, \alpha n)$-source for some constant $\alpha > 0$ (i.e., X has linear min-entropy); and Y is an (n,k) source, where $k \geq \mathrm{polylog}(n)$.

We first define an extractor for a block source and an independent general source, as in [RZ08, BRSW06]:

$$\mathsf{BExt} : \{0,1\}^{n_1} \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$$

- **Ingredients**:
  - $\mathsf{Zuc} : \{0,1\}^{\alpha n/2} \to \{0,1\}^{c \times \ell}$ a function as in Theorem A.16, where $c = \mathrm{poly}(1/\alpha)$ and $\ell = \frac{n}{\mathrm{poly}(1/\alpha)}$.
  - $\mathsf{Raz} : \{0,1\}^\ell \times \{0,1\}^n \to \{0,1\}^s$ a strong 2-source extractor as in Theorem A.23, where one source is a $(\ell, 0.9\ell)$-source and the other is an $(n, 0.9k)$-source (where k is the min-entropy of Y).

- $\mathsf{BasicExt} : \{0,1\}^n \times \{0,1\}^{s^\gamma \times s} \to \{0,1\}^d$ the extractor as in Theorem A.22, where $s = \mathrm{polylog}(n)$.

- $\mathsf{BExt}(x_1, x, y)$:

  1. Compute $v = \mathrm{Zuc}(x_1)$.
  2. Apply the strong 2-source extractor Raz to each row of v and y. Denote the resulting matrix by sr. Note each row of sr is of length $s = \mathrm{polylog}(n)$.
  3. Output $\mathsf{BasicExt}(x, sr)$

Next we use this block source extractor to extract

$$\mathsf{TExt} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$$

- **Ingredients**:
  - $\mathsf{RExt} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ an extractor for entropy $0.9k$ as in Theorem A.19, where $\epsilon = \frac{1}{\mathrm{poly}(n^{\log n})}$ and $d = O(\log(n/\epsilon)) = \mathrm{polylog}(n)$.
  - f, a one way permutation for $0.3k$-sources

- $\mathsf{TExt}(x, y)$:

  Partition x, into $t = \frac{4}{\alpha}$ equally sized parts[3] $x = (x_1, \ldots, x_t)$.

  For every $i = 1, \ldots, t$,

  1. Compute $r_i = \mathsf{BExt}(x_i, x, f^{(i)}(y))$, and assume without loss of generality that r_i is a d length string where $d = \mathrm{polylog}(n)$ is the seed length for the extractor RExt.
  2. Compute $z_i = \mathsf{RExt}(f^{(i)}(y), r_i)$, and assume without loss of generality that z_i is of length $m \leq \mathrm{polylog}(n)$.

  Output the bitwise parity $z = \oplus_{i=1}^t z_i$.

B.2 Analysis of the Extractor

We have the following theorem:

Theorem B.1. *Fix a constant $\alpha > 0$ and parameters $t = \frac{4}{\alpha}$ and $k \geq n^{\Omega(1)}$. Assume that there exists a permutation $f : \{0,1\}^n \to \{0,1\}^n$ such that for any $(n, 0.3k)$-source Y, any non-uniform adversary that runs in time $\mathrm{poly}(n^{\log n})$ can invert $f(Y)$ with only negligible probability. Then* $\mathsf{TExt} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$ *described above is a computational 2-source extractor such that for any $(n, \alpha n)$-source X, and any (n, k)-source Y that is independent of X,*

$$(\mathsf{TExt}(X, Y), X) \approx (U_m, X)$$

[3] For the sake of simplicity, we assume t is an integer.

Remark.

- Rather than proving Theorem B.1, we prove the following (stronger) statement:
$$(\mathsf{TExt}(X,Y), X, h(X), f^{(t+1)}(Y)) \approx (U_m, X, h(X), f^{(t+1)}(Y)), \tag{2}$$
where h is any deterministic function (not necessarily computable in polynomial time) on $\{0,1\}^n$. The reason is that we need this stronger variant for our network extractor protocol in Section **??**.

- If we let $m = O(\log n)$, then the theorem implies that in particular,
$$|\mathsf{TExt}(X,Y) - U_m| = \mathrm{negl}(n)$$
.

To prove Equation (2) we first prove the following lemma(in the analysis we use capital letters to denote the corresponding strings viewed as random variables).

Lemma B.2. *Divide X into $X = (X_1, \ldots, X_t)$ as in the construction of* TExt*, and let $Z = \mathsf{TExt}(X,Y)$. Suppose there exists $g \in [t]$ such that*

- *$X_1, \ldots, X_{g-1}$ are fixed.*
- *$H_\infty(X_g) \geq \frac{\alpha^2}{6}$.*
- *$H_\infty(X|X_g) \geq \frac{\alpha^2}{6}$.*

Then
$$(Z, X, h(X), f^{(t+1)}(Y)) \approx (U_m, X, h(X), f^{(t+1)}(Y))$$

Proof of Lemma B.2. Let sr_i denote the string sr computed in $\mathsf{BExt}(x_i, x, f^{(i)}(Y))$. Fix
$$(sr_1, \ldots, sr_{g-1}) \leftarrow (SR_1, \ldots, SR_{g-1})$$
$$(r_1, \ldots, r_{g-1}) \leftarrow (R_1, \ldots, R_{g-1})$$
and
$$(z_1, \ldots, z_{g-1}) \leftarrow (Z_1, \ldots, Z_{g-1}).$$

For any random variable Z, we denote by Z' the random variable Z conditioned on these fixings.

Let TYPICAL denote the event that conditioned on these fixings, the following conditions are satisfied:

- X' and Y' are independent
- $H_\infty(Y') \geq k - \mathrm{polylog}(n)$
- $H_\infty(X'_g) \geq \frac{\alpha^2}{8} n$
- With probability $1 - \mathrm{negl}(n)$ over $(x'_g \leftarrow X'_g)$, $H_\infty(X'|X'_g = x'_g) \geq \frac{\alpha^2}{8} n$

Claim B.3.
$$\Pr[\mathrm{TYPICAL}] = 1 - \mathrm{negl}(n).$$

Proof of Claim B.3. Since $X_1, \dots, X_{g-1}$ are fixed, $(SR_1, \dots, SR_{g-1})$ is a deterministic function of Y. Thus, conditioning on $(sr_1, \dots, sr_{g-1}) \leftarrow (SR_1, \dots, SR_{g-1})$, X and Y are still independent. Moreover, since each sr_i has size cs, the total size of $(sr_1, \dots, sr_{g-1})$ is bounded by $tcs = \mathrm{polylog}(n)$. Thus, by Lemma A.12, with probability $1 - \mathrm{negl}(n)$ over these fixings, Y has min-entropy $k - \mathrm{polylog}(n)$ (let $\epsilon = 2^{-\log^2 n}$ in the lemma).

Next, we further condition on $(r_1, \dots, r_{g-1}) \leftarrow (R_1, \dots, R_{g-1})$. Note that now $(R_1, \dots, R_{g-1})$ is a deterministic function of X. Thus, conditioned on this fixing, X and Y are still independent. Moreover, since each r_i is of size $\mathrm{polylog}(n)$, the total size of $(r_1, \dots, r_{g-1})$ is bounded by $t|r_i| = \mathrm{polylog}(n)$. Thus, by Lemma A.12, with probability $1 - \mathrm{negl}(n)$ over these fixings, X_g has min-entropy $H_\infty(X_g) - \mathrm{polylog}(n) > \frac{\alpha^2}{8}n$ (let $\epsilon = 2^{-\log^2 n}$ in the lemma). Next, note that conditioned on any fixing of $x_g \in \mathsf{Supp}(X_g)$, we have $H_\infty(X) \geq \frac{\alpha^2}{6}n$, and with probability $1 - \mathrm{negl}(n)$ over the further fixings of $(R_1, \dots, R_{g-1})$, $H_\infty(X) > \frac{\alpha^2}{8}n$. Thus we have

$$\Pr_{X_g, R_1, \dots, R_{g-1}}[H_\infty(X' | X'_g = x'_g) > \frac{\alpha^2}{8}n] \geq 1 - \epsilon_1,$$

where $\epsilon_1 = \mathrm{negl}(n)$.

Now a standard averaging argument shows that, with probability at least $1 - \sqrt{\epsilon_1}$ over the fixings of $(R_1, \dots, R_{g-1})$,

$$\Pr_{x'_g \leftarrow X'_g}[H_\infty(X' | X'_g = x'_g) > \frac{\alpha^2}{8}n] \geq 1 - \sqrt{\epsilon_1}.$$

Note $\epsilon_1 = \mathrm{negl}(n)$, thus $\sqrt{\epsilon_1} = \mathrm{negl}(n)$.

Finally, we further condition on $(z_1, \dots, z_{g-1}) \leftarrow (Z_1, \dots, Z_{g-1})$. Note that now $(Z_1, \dots, Z_{g-1})$ is a deterministic function of Y. Thus, conditioned on this fixing, X and Y are still independent. Moreover, since each z_i is of size $\mathrm{polylog}(n)$, the total size of $(z_1, \dots, z_{g-1})$ is bounded by $t|z_i| = \mathrm{polylog}(n)$. Thus, by Lemma A.12, with probability $1 - \mathrm{negl}(n)$ over these fixings, Y has min-entropy $k - \mathrm{polylog}(n) - \mathrm{polylog}(n) = k - \mathrm{polylog}(n)$ (let $\epsilon = 2^{-\log^2 n}$ in the lemma).

The probability that all of the above happen is at least $1 - \mathrm{negl}(n) - \mathrm{negl}(n) - \mathrm{negl}(n) = 1 - \mathrm{negl}(n)$. ■

Now fix

$$(x'_g, sr'_g) \leftarrow (X'_g, SR'_g).$$

For every random variable Z', denote by

$$Z'' = Z' | (X'_g = x'_g, SR'_g = sr'_g).$$

Let TYPICAL2 denote the event that conditioned on all the above fixings, the following holds:

- X'' and Y'' are independent, R''_g is a deterministic function of X''
- $H_\infty(Y'') \geq 0.9k$
- $(R''_g, Y'') \equiv (U_d, Y'')$

Claim B.4. *If* TYPICAL *holds, then*

$$\Pr[\text{TYPICAL2}] = 1 - \mathrm{negl}(n)$$

Proof of Claim B.4. First note that when TYPICAL holds, X' and Y' are independent, and $H_\infty(X'_g) \geq \frac{\alpha^2}{8}n$. This means X'_g has min-entropy rate $\geq \frac{\alpha}{2}$. Therefore by Theorem A.16 M'_g is $2^{-\Omega(n)}$-close to a $(c \times \ell)0.9\ell$-somewhere random source. Theorem A.23 implies that there exists a somewhere-random source SR with c rows, each row of length s, s.t.

$$|(M'_g, SR'_g) - (M'_g, SR)| = \mathrm{negl}(n).$$

A standard averaging argument shows that with probability $1 - \mathrm{negl}(n)$ over the fixing of M'_g (and thus X'_g), we still have

$$|SR'_g - SR| = \mathrm{negl}(n).$$

Moreover, X'_g (and thus M'_g) is a deterministic function of X', thus conditioned on the fixing of X'_g (and thus M'_g), X' and Y' are still independent. Note once conditioned on M'_g, SR'_g is a deterministic function of Y', and is thus independent of X'. Also, with probability $1 - \mathrm{negl}(n)$ over the fixing of X'_g, $H_\infty(X') \geq \frac{\alpha^2}{8}n$. The probability that both these two events happen is $1 - \mathrm{negl}(n)$, and when this happens, Theorem A.22 implies that

$$|(SR'_g, R_g) - (SR'_g, U_d)| < 2^{-n^{\Omega(1)}} + \mathrm{negl}(n) = \mathrm{negl}(n).$$

Since this happens with probability $1 - \mathrm{negl}(n)$, we actually have that

$$|(SR'_g, R_g) - (SR'_g, U_d)| = \mathrm{negl}(n).$$

Again, by a standard averaging argument, with probability $1 - \mathrm{negl}(n)$ over the fixing of SR'_g, we still have

$$|R_g - U_d| = \mathrm{negl}(n).$$

Note since SR'_g is a deterministic function of Y', conditioning on it still leaves X' and Y' independent. Moreover, since the size of sr_g is small, the same argument in the proof of Claim B.3 implies that with probability $1 - \mathrm{negl}(n)$ over the fixings of SR'_g, $H_\infty(Y') \geq k - \mathrm{polylog}(n) - \mathrm{polylog}(n) > 0.9k$. Finally, conditioned on the fixing of SR'_g, R_g is a deterministic function of X', and is thus independent of Y'. Note $|R_g - U_d| = \mathrm{negl}(n)$, therefore

$$(R_g, Y'') \equiv (U_d, Y'')$$

The probability that all of the above are satisfied is $1 - \mathrm{negl}(n)$. ■

Next we prove the following claim.

Claim B.5. *If both* TYPICAL *and* TYPICAL2 *hold, then*

$$(\oplus_{i=g}^{t} Z''_i, X'', h(X''), f^{t+1}(Y'')) \approx (U_m, X'', h(X''), f^{t+1}(Y''))$$

Proof of Claim B.5. Assume for the sake of contradiction that there exists a non-uniform PPT adversary $\mathcal{A}_1$ and a polynomial q such that for infinitely many n's

$$\left|\Pr[\mathcal{A}_1(\oplus_{i=g}^{t} Z''_i, X'', h(X''), f^{t+1}(Y'')) = 1] - \Pr[\mathcal{A}_1(U_m, X'', h(X''), f^{t+1}(Y'')) = 1\right| \geq \frac{1}{q(n)}.$$

Since $Z''_{g+1}, \cdots, Z''_t$ and $f^{t+1}(Y'')$ can be computed from $(X'', f^{g+1}(Y''))$ in polynomial time, there exists another non-uniform PPT adversary $\mathcal{A}_2$ such that

$$\left|\Pr[\mathcal{A}_2(Z''_g, R''_g, X'', h(X''), f^{g+1}(Y'')) = 1] - \Pr[\mathcal{A}_2(U_m, R''_g, X'', h(X''), f^{g+1}(Y'')) = 1\right| \geq \frac{1}{q(n)}.$$

Recall that

$$Z''_g = \mathsf{RExt}(f^{(g)}(Y''), R''_g)$$

and $f^{(g)}(Y'')$ is a deterministic function of $f^{(g+1)}(Y'')$(though not computable in polynomial time). Thus Z''_g is a deterministic function of $f^{(g+1)}(Y'')$ and R''_g. Next note that R''_g is a deterministic function of X'', and X'', Y'', U_m are independent. Thus Lemma A.20 implies that there exists another non-uniform adversary $\mathcal{A}_3$ that runs in time $2^d \cdot n \cdot \text{poly}(n) = \text{poly}(n, \frac{1}{\epsilon}) \cdot \text{poly}(n) = \text{poly}(n^{\log n})$ such that

$$\left|\Pr[\mathcal{A}_3(Z''_g, R''_g, f^{g+1}(Y'')) = 1] - \Pr[\mathcal{A}_3(U_m, R''_g, f^{g+1}(Y'')) = 1\right| \geq \frac{1}{q(n)}.$$

Note that the fact that TYPICAL2 holds implies that $(R''_g, Y'') \equiv (U_d, Y'')$. This, together with Proposition A.6 implies that

$$\left|\Pr[\mathcal{A}_3(Z''_g, R, f^{g+1}(Y'')) = 1] - \Pr[\mathcal{A}_3(U_m, R, f^{g+1}(Y'')) = 1\right| \geq \frac{1}{q(n)} - \text{negl}(n) > \frac{1}{2q(n)},$$

where R is the uniform distribution on $\{0,1\}^d$ and is independent of Y''.

Now note $\frac{1}{2q(n)} > 2\epsilon$ since $\epsilon = \frac{1}{\text{poly}(n^{\log n})}$, and $f^{(g)}(Y'')$ has min-entropy $0.9k$ since f is a permutation. Thus Theorem A.19 implies that there exists another non-uniform adversary $\mathcal{A}_4$ that runs in time $\text{poly}(n, \frac{1}{\epsilon}) \cdot \text{poly}(n^{\log n}) = \text{poly}(n^{\log n})$ and an $(n, 0.3k)$-source $\bar{Y}$ such that $\mathcal{A}_4$ inverts $f(\bar{Y})$ with probability at least $\frac{1}{16q(n)}$. This contradicts our assumption on f. $\blacksquare$

Now, since the event that both TYPICAL and TYPICAL2 hold happens with probability $1 - \text{negl}(n)$, by Lemma A.7 we have

$$(\oplus_{i=g}^{t} Z_i, \{Z_i\}_{i \in [g-1]}, X, h(X), f^{t+1}(Y)) \approx (U_m, \{Z_i\}_{i \in [g-1]}, X, h(X), f^{t+1}(Y)).$$

Note that $Z = \oplus_{i=1}^{t} Z_i$, thus

$$(Z, X, h(X), f^{t+1}(Y)) \approx (U_m, X, h(X), f^{t+1}(Y)).$$

This proves the lemma. $\blacksquare$

proof of Theorem B.1. Since we divide X into $t = \frac{4}{\alpha}$ blocks, Lemma A.11 says that X is $2^{-n^{\Omega(1)}}$-close to being a convex combination of $\{X^j\}_{j \in J}$ such that for every $j \in J$, X_j satisfies the conditions in Lemma B.2. For every $j \in J$, let $Z^j = \mathsf{TExt}(X^j, Y)$, then

$$(Z^j, X^j, h(X^j), f^{(t+1)}(Y)) \approx (U_m, X^j, h(X^j), f^{(t+1)}(Y)).$$

Thus, by Lemma A.8, the theorem holds. $\blacksquare$

The computational two source extractor described above outputs random bits that are computationally indistinguishable from being uniform, while assuming the existence of a one-way permutation f s.t. for any $(n, 0.3k)$ source X, any non-uniform adversary that runs in time $\mathrm{poly}(n^{\log n})$ can only invert $f(X)$ with negligible probability. Note that the running time of the adversary is slightly super-polynomial. However, even if we only assume that any polynomial time adversary can only invert $f(X)$ with negligible error, we can still get a two-source extractor, but the error will only be polynomially small.

Theorem B.6. *Let $\epsilon = \frac{1}{\mathrm{poly}(n)}$ and $m = O(\log n)$ in the construction of TExt. Keep all the other parameters the same. Assume that there exists a permutation $f : \{0,1\}^n \to \{0,1\}^n$ such that for any $(n, 0.3k)$-source Y, any non-uniform adversary that runs in time $\mathrm{poly}(n)$ can invert $f(Y)$ with only negligible probability. Then TExt is a 2-source extractor such that for any $(n, \alpha n)$-source X, and any (n, k)-source Y that is independent of X,*

$$|\mathsf{TExt}(X, Y) - U_m| < 3\epsilon.$$

Proof Sketch. The proof basically follows the same steps in the proof of Theorem B.1. We first prove that for a source X that satisfies the conditions of Lemma B.2, and $Z = \mathsf{TExt}(X, Y)$, we have

$$|Z - U_m| < 2.9\epsilon. \tag{3}$$

To this end, we prove that when both TYPICAL and TYPICAL2 hold, we have

$$|Z'' - U_m| < 2.5\epsilon. \tag{4}$$

Assume for the sake of contradiction that $|Z'' - U_m| \geq 2.5\epsilon$. Since $m = O(\log n)$, there exists a non-uniform PPT adversary $\mathcal{A}$(simply check all the 2^m strings) s.t.

$$\left|\Pr[\mathcal{A}(Z'') = 1] - \Pr[\mathcal{A}(U_m) = 1]\right| \geq 2.5\epsilon$$

Note $Z'' = \oplus_{i=1}^t Z_i''$. Since $Z_1, \ldots, Z_{g-1}$ are fixed, and $Z_{g+1}'', \cdots, Z_t''$ can be computed from $(X'', f^{g+1}(Y''))$ in polynomial time, there exists another non-uniform PPT adversary $\mathcal{A}_1$ such that

$$\left|\Pr[\mathcal{A}_1(Z_g'', R_g'', X'', f^{g+1}(Y'')) = 1] - \Pr[\mathcal{A}_1(U_m, R_g'', X'', f^{g+1}(Y'')) = 1]\right| \geq 2.5\epsilon$$

Recall that $Z_g'' = \mathsf{RExt}(f^{(g)}(Y''), R_g'')$, thus Lemma A.20 implies that there exists another non-uniform adversary $\mathcal{A}_2$ that runs in time $2^d \cdot n \cdot \mathrm{poly}(n) = \mathrm{poly}(n, \frac{1}{\epsilon}) \cdot \mathrm{poly}(n) = \mathrm{poly}(n)$ such that

$$\left|\Pr[\mathcal{A}_2(Z_g'', R_g'', f^{g+1}(Y'')) = 1] - \Pr[\mathcal{A}_2(U_m, R_g'', f^{g+1}(Y'')) = 1\right| \geq 2.5\epsilon.$$

Note that the fact that TYPICAL2 holds implies that $(R_g'', Y'') \equiv (U_d, Y'')$. This, together with Proposition A.6 implies that

$$\left|\Pr[\mathcal{A}_2(Z_g'', R, f^{g+1}(Y'')) = 1] - \Pr[\mathcal{A}_2(U_m, R, f^{g+1}(Y'')) = 1\right| \geq 2.5\epsilon - \mathrm{negl}(n) > 2\epsilon,$$

where R is the uniform distribution on $\{0,1\}^d$ and is independent of Y''. Note $f^{(g)}(Y'')$ has min-entropy $0.9k$ since f is a permutation. Thus Theorem A.19 implies that there exists another non-uniform adversary $\mathcal{A}_3$ that runs in time $\text{poly}(n, \frac{1}{\epsilon}) \cdot \text{poly}(n^{\log n}) = \text{poly}(n^{\log n})$ and an $(n, 0.3k)$-source $\bar{Y}$ such that $\mathcal{A}_3$ inverts $f(\bar{Y})$ with probability at least $\epsilon/4$. This contradicts our assumption on f.

Thus Equation 4 does hold. Since the event that both TYPICAL and TYPICAL2 hold happens with probability $1 - \text{negl}(n)$, Equation 3 holds. Now by Lemma A.11, X is $2^{-n^{\Omega(1)}}$-close to being a convex combination of $\{X^j\}_{j \in J}$ such that for every $j \in J$, X_j satisfies the conditions in Lemma B.2. Thus the theorem holds. ■

C Computational Network Extractors for Linear Min-Entropy

We next use our 2-source extractor to construct a computational network extractor protocol. In order to formally define the notion of network extractors, we need some notation. We denote the input of player i by $x_i \in \{0,1\}^n$; each x_i is assumed to be sampled from an independent (n, k)-source. We denote the output of player i by $z_i \in \{0,1\}^m$, which is supposedly a random looking string. We denote by b the concatenation of all the messages that were sent during the protocol. Capital letters such as Z_i and B denote these strings viewed as random variables.

Definition C.1. A protocol for p processors is a (t, g) **computational network extractor** for min-entropy k if for any independent (n, k)-sources and any choice of t faulty processors, after running the protocol there are g honest processors $\mathcal{G} = \{i_1, \ldots, i_g\}$ such that

$$\{B, (X_i)_{i \notin \mathcal{G}}, (Z_i)_{i \in \mathcal{G}}\}_{n \in \mathbb{N}} \approx \{B, (X_i)_{i \notin \mathcal{G}}, U_{gm}\}_{n \in \mathbb{N}}$$

where U_{gm} is the uniform distribution on gm bits, independent of B and $(X_i)_{i \notin \mathcal{G}}$, and where $\approx$ denotes computational indistinguishability.

Remark. Note that it must be the case that $g \leq p - t$, since there are $p - t$ honest players (and we cannot hope that the output of a faulty player will be indistinguishable from random). Ideally, we would like to design network extractors where $g = p - t$, which means that the joint output of *all* the honest players is computationally indistinguishable from random, even given the entire transcript and all the sources of the dishonest players.

We next show how to use our computational 2-source extractor (Theorem B.1), to construct such a computational network extractor protocol for any number of players p as long as at least 2 of them are honest, and each honest processor starts with an independent $(n, \alpha n)$ source. In order to use our 2-source extractor, we need to make the following assumption.

We first note that Theorem A.19 implies the following corollary:

Corollary C.2. *If f is a one way permutation for $0.3\alpha n$-sources, then there is an efficiently computable function* RExt *as in Theorem A.19, with parameters $\epsilon = \frac{1}{\text{poly}(n^{\log n})}$ and $d = O(\log(n/\epsilon)) =$* $\text{polylog}(n)$*, such that for any $(n, 0.9\alpha n)$ source X,*

$$(\mathsf{RExt}(X, U_d), f(X), U_d) \approx (U_m, f(X), U_d).$$

Proof of Corollary C.2. Assume for the sake of contradiction that there exists a non-uniform PPT adversary $\mathcal{A}$ and a polynomial q such that for infinitely many n's,

$$|\Pr[\mathcal{A}(\mathsf{RExt}(X, U_d), f(X), U_d) = 1] - \Pr[\mathcal{A}(U_m, f(X), U_d) = 1]| \geq \frac{1}{q(n)}.$$

Note $\frac{1}{q(n)} > 2\epsilon$. Thus, by Theorem A.19, there exists another non-uniform adversary $\mathcal{B}$ that runs in time $\mathrm{poly}(n, 1/\epsilon) \cdot \mathit{Time}(\mathcal{A}) = \mathrm{poly}(n^{\log n})$ and an $(n, 0.3\alpha n)$ source $\bar{X}$ such that $\mathcal{B}$ inverts $f(\bar{X})$ with probability at least $\frac{1}{8q(n)}$. This contradicts the fact that f is one-way. ■

C.0.1 The Protocol

- **Parameters.**
  - Constant $\alpha > 0$, where αn is the min-entropy of each of the input sources X_i.
  - Parameters d, ϵ as in Corollary C.2 and $m = \mathrm{polylog}(n)$. Note $d = \mathrm{polylog}(n)$ and $\epsilon = \mathrm{negl}(n)$.
  - $t = \lceil \frac{4}{0.9\alpha} \rceil$.
- **Ingredients.**
  - The 2-source extractor $\mathsf{TExt} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$ from Theorem B.1.
  - $\mathsf{RExt} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ a $(0.9\alpha n, \epsilon)$-extractor as in Theorem A.19.
  - $f : \{0,1\}^n \to \{0,1\}^n$ a one way permutation for $0.3\alpha n$-sources. Let $g = f^{(t+1)}$.
- **The protocol.** The protocol proceeds in two phases.
  - **Phase 1.** The first phase of the protocol proceeds in p rounds (where p is the number of players). In round $j \in [p]$ the players do the following.
    1. The j'th player sends $g^{(2j)}(X_j)$ to all other players, where $g = f^{(t+1)}$ is described above.
    2. Each player i computes $R_i^j = \mathsf{TExt}(g^{(2j)}(X_j), g^{(2j)}(X_i))$. Note $|R_i^j| = m = \mathrm{polylog}(n)$.

    At the end of the p'th round each player i computes $R_i = \oplus_{j=1}^{p} R_i^j$.

    We show that at the end of this phase all the honest players, except for the first one, end up with private randomness. In order to ensure that *all* the honest players, *including the first one*, end up with private randomness, we proceed to the second phase.
  - **Phase 2.** Each player i partitions R_i into two equal parts $R_i = (V_i, W_i)$. All players engage in a secure multiparty computation to compute $V = \oplus_{i=1}^{p} V_i$[4], where each player i uses W_i as its internal randomness.

    Finally, each player i outputs $Z_i = \mathsf{RExt}(g^{(2i-1)}(X_i), V)$.

Theorem C.3. *For any $p \geq 2$, any $t \leq p - 2$, and any $n \geq p^{1+\gamma}$ (for some constant $\gamma > 0$), the protocol described above is a $(t, p-t)$-computational network extractor protocol.*

[4] Here, if the adversary aborts the protocol, we simply discard the aborting party and restart the secure computation with fresh V_i's

Proof. Fix any p and any t such that $p-t \geq 2$. Let $\mathcal{G} \subseteq [p]$ denote the set of all honest players. Let j denote the GOOD round, where the first honest player sends its source; i.e., $\mathcal{G} \subseteq \{j, \ldots, p\}$. We assume without loss of generality that the PPT adversary (who controls all the malicious players) is deterministic. Thus, it suffices to prove that

$$\{B, (Z_i)_{i \in \mathcal{G}}\} \approx \{B, U_{gm}\}.$$

The proof proceeds in two parts. In the first part we prove that all the R_i's of all the honest players, except player j, appear to be independent and uniformly distributed, even conditioned on the entire transcript of phase 1. Actually ,we prove the following stronger statement:

$$\{X_j, (g^{(2j+1)}(X_i))_{i \in \mathcal{G} \setminus \{j\}}, (R_i)_{i \in \mathcal{G} \setminus \{j\}}\} \approx \{X_j, (g^{(2j+1)}(X_i))_{i \in \mathcal{G} \setminus \{j\}}, U_{(g-1)m}\} \tag{5}$$

Note that this statement is indeed stronger since for every $i \in \mathcal{G} \setminus \{j\}$ it holds that $i > j$, which implies that $2i > 2j+1$, and therefore it is easy to compute $g^{(2i)}(X_i)$ from $g^{(2j+1)}(X_i)$.

In the second part we use Equation (5) to prove that indeed

$$\{(B, (Z_i)_{i \in \mathcal{G}}\} \approx \{(B, U_{gm}\}.$$

Part 1. Assume for the sake of contradiction that there exists a non-uniform PPT adversary $\mathcal{A}$ and a polynomial q such that for infinitely many n's

$$\begin{aligned} \big| \Pr[\mathcal{A}(X_j, (g^{(2j+1)}(X_i))_{i \in \mathcal{G} \setminus \{j\}}, (R_i)_{i \in \mathcal{G} \setminus \{j\}}) = 1] - \\ \Pr[\mathcal{A}(X_j, (g^{(2j+1)}(X_i))_{i \in \mathcal{G} \setminus \{j\}}, U_{(g-1)m}) = 1] \big| \geq \frac{1}{q(n)}. \end{aligned}$$

Let B_{j-1} denote the transcript until round $j-1$. Fix any tuple

$$\{b_{j-1}, (r_i^1, \ldots, r_i^{j-1})_{i \in \mathcal{G}}\} \leftarrow \{B_{j-1}, (R_i^1, \ldots, R_i^{j-1})_{i \in \mathcal{G}}\}.$$

Denote any random variable Y, conditioned on the above tuple, by Y'. We say that the tuple above is BAD if the following properties are satisfied.

1. There are infinitely many n's for which

$$\begin{aligned} \big| \Pr[\mathcal{A}(X'_j, (g^{(2j+1)}(X'_i))_{i \in \mathcal{G} \setminus \{j\}}, (R'_i)_{i \in \mathcal{G} \setminus \{j\}}) = 1] - \\ \Pr[\mathcal{A}(X'_j, (g^{(2j+1)}(X'_i))_{i \in \mathcal{G} \setminus \{j\}}, U_{(g-1)m}) = 1] \big| \geq \frac{1}{2q(n)}. \end{aligned}$$

2. For every $i \in \mathcal{G}$, the random variable X'_i has min-entropy $0.9\alpha n$ and $\{X'_i\}_i \in \mathcal{G}$ are independent random variables.

Claim C.4.

$$\Pr[\{b_{j-1}, (r_i^1, \ldots, r_i^{j-1})_{i \in \mathcal{G}}\} \text{ is BAD}] \geq \frac{1}{3q(n)}.$$

Proof of Claim C.4. A standard probabilistic argument shows that the probability that a random tuple

$$\{b_{j-1},(r_i^1,\ldots,r_i^{j-1})_{i\in\mathcal{G}}\}\leftarrow\{B_{j-1},(R_i^1,\ldots,R_i^{j-1})_{i\in\mathcal{G}}\}$$

satisfies the first property with probability at least $\frac{1}{2q(n)}$.
The fact that the random variables $\{X_i'\}_{i\in\mathcal{G}}$ remain independent can be seen by induction on the number of rounds. Moreover, since all the R_i^j's are of size only $\mathrm{polylog}(n)$ and $n\geq p^{1+\gamma}$, by Lemma A.12 with probability $1-\mathrm{negl}(n)$ over the fixings, every X_i' has min-entropy at least $0.9\alpha n$ (take $\epsilon=2^{-\log^2 n}$ in that lemma).

Thus the probability that a random tuple is BAD is at least $\frac{1}{2q(n)}-\mathrm{negl}(n)\geq\frac{1}{3q(n)}$

■

Fix any BAD tuple

$$\{b_{j-1},(r_i^1,\ldots,r_i^{j-1})_{i\in\mathcal{G}}\}\leftarrow\{B_{j-1},(R_i^1,\ldots,R_i^{j-1})_{i\in\mathcal{G}}\}.$$

Note that $R_i=\oplus_{j=1}^{p}R_i^j$. Since $(R_i^1,\ldots,R_i^{j-1})_{i\in\mathcal{G}}$ are fixed, and $(R_i^{j+1},\ldots,R_i^p)_{i\in\mathcal{G}}$ can be computed in polynomial time from $(g^{(2j+1)}(X_i'))_{i\in\mathcal{G}\setminus\{j\}}$, there exists another non-uniform PPT adversary $\mathcal{A}_1$ such that

$$\begin{aligned}&\big|\Pr[\mathcal{A}_1(X_j',(g^{(2j+1)}(X_i'))_{i\in\mathcal{G}\setminus\{j\}},(R_i'^j)_{i\in\mathcal{G}\setminus\{j\}})=1]-\\&\Pr[\mathcal{A}_1(X_j',(g^{(2j+1)}(X_i'))_{i\in\mathcal{G}\setminus\{j\}},U_{(g-1)m})=1]\big|\geq\frac{1}{2q(n)}.\end{aligned}$$

A standard hybrid argument shows that there exists $\ell\in\mathcal{G}\setminus\{j\}$ (note there are at least 2 honest players) s.t.

$$\begin{aligned}&\big|\Pr[\mathcal{A}_1(X_j',(g^{(2j+1)}(X_i'))_{i\in\mathcal{G}\setminus\{j\}},(R_i'^j)_{i\in\mathcal{G}\setminus\{j,\ell\}},R_\ell'^j)=1]-\\&\Pr[\mathcal{A}_1(X_j',(g^{(2j+1)}(X_i'))_{i\in\mathcal{G}\setminus\{j\}},(R_i'^j)_{i\in\mathcal{G}\setminus\{j,\ell\}},U_m)=1]\big|\geq\frac{1}{2p\cdot q(n)}.\end{aligned}$$

Since the random variables $\{X_i'\}_{i\in\mathcal{G}}$ remain independent, there is a fixing of $(X_i')_{i\in\mathcal{G}\setminus\{j,\ell\}}$ that preserves the distinguishing probability. Note after this fixing, $(R_i'^j)_{i\in\mathcal{G}\setminus\{j,\ell\}}$ is a deterministic function of X_j'. Thus, there exists another non-uniform PPT adversary $\mathcal{B}$, that has all the fixings hardwired into it, s.t.

$$\big|\Pr[\mathcal{B}(X_j',g^{(2j+1)}(X_\ell'),R_\ell'^j)=1]-\Pr[\mathcal{B}(X_j',g^{(2j+1)}(X_\ell'),U_m)=1]\big|\geq\frac{1}{2p\cdot q(n)}.\tag{6}$$

Note X_j' and X_ℓ' are independent, both have min-entropy at least $0.9\alpha n$ and X_j' is a deterministic function of $g^{(2j)}(X_j')$. Moreover, recall that

$$R_\ell'^j=\mathsf{TExt}(g^{(2j)}(X_j'),g^{(2j)}(X_\ell')).$$

Thus, according to Theorem B.1 (or more precisely, Equation (2), where $h=g^{(-2j)}$),

$$(R_\ell'^j,X_j',f^{(t+1)}(g^{(2j)}(X_\ell')))\approx(U_m,X_j',f^{(t+1)}(g^{(2j)}(X_\ell'))).$$

Note $f^{(t+1)}(g^{(2j)}(X'_\ell)) = g^{(2j+1)}(X'_\ell))$, thus

$$(R'^j_\ell, X'_j, g^{(2j+1)}(X'_\ell)) \approx (U_m, X'_j, g^{(2j+1)}(X'_\ell)),$$

which contradicts Equation (6). Therefore, we conclude that indeed Equation (5) holds. Namely,

$$\{X_j, (g^{(2j+1)}(X_i))_{i\in\mathcal{G}\setminus\{j\}}, (R_i)_{i\in\mathcal{G}\setminus\{j\}}\} \approx \{X_j, (g^{(2j+1)}(X_i))_{i\in\mathcal{G}\setminus\{j\}}, U_{(g-1)m}\}$$

Part 2. We now use Equation (5) to prove our final statement

$$\{B, (Z_i)_{i\in\mathcal{G}}\} \approx \{B, U_{gm}\}.$$

We parse $B = (B_1, B_2)$ where B_1 denotes the transcript of Phase 1, and B_2 denotes the transcript of Phase 2. Thus, we need to prove that

$$\{B_1, B_2, (Z_i)_{i\in\mathcal{G}}\} \approx \{B_1, B_2, U_{gm}\}.$$

Recall that Phase 2 consists only of a secure multiparty computation of $V = \oplus V_i$. By the definition of secure multiparty computation, all the transcript of the second phase can be simulated given V, given all the sources of the malicious players $(X_i)_{i\notin\mathcal{G}}$, and given R_j. The reason we need to give also R_j is that during this secure computation player P_j (who is the first honest player), may not have private randomness, and therefore we think of this player as being malicious. Thus, it suffices to prove that

$$\{B_1, R_j, V, (Z_i)_{i\in\mathcal{G}}\} \approx \{B_1, R_j, V, U_{gm}\}. \tag{7}$$

We first notice that for every $i \in \mathcal{G} - \{j\}$,

$$2i - 1 \geq 2j + 1.$$

This follows from our assumption that j is the first honest player, and thus for every $i \in \mathcal{G}\setminus\{j\}$ it holds that $i \geq j+1$.

This, together with Equation (5), implies that

$$\{X_j, (g^{(2i-1)}(X_i))_{i\in\mathcal{G}\setminus\{j\}}, (R_i)_{i\in\mathcal{G}\setminus\{j\}}\} \approx \{X_j, (g^{(2i-1)}(X_i))_{i\in\mathcal{G}\setminus\{j\}}, U_{(g-1)m}\},$$

which in turn implies that

$$\{R_j, (g^{(2i-1)}(X_i))_{i\in\mathcal{G}}, V\} \approx \{R_j, (g^{(2i-1)}(X_i))_{i\in\mathcal{G}}, U\}. \tag{8}$$

Remark C.5. Here we would like to think of U as being uniform, but the complete proof for this equation is somewhat involved: the adversary can choose to abort the protocol in the secure computation. What is true is that the indistinguishability holds with U being chosen adversarially from a set of t uniformly sampled strings. Since the number of aborts is at most the number of dishonest players (which is bounded by $\mathrm{poly}(n)$), any adversary that can distinguish the two sides with a truly uniform U can also succeed in the case that U is distributed as we described.

Next, notice that it is easy to simulate the transcript B_1 given

$$\left((X_i)_{i\notin\mathcal{G}}, (g^{(2i)}(X_i))_{i\in\mathcal{G}}\right).$$

Therefore, to prove Equation (7) it suffices to prove that

$$\{R_j, (g^{(2i)}(X_i))_{i\in\mathcal{G}}, V, (\mathsf{RExt}(g^{(2i-1)}(X_i), V))_{i\in\mathcal{G}}\} \approx \{R_j, (g^{(2i)}(X_i))_{i\in\mathcal{G}}, V, U_{gm}\}.$$

This is immediately implied from Equation (8), from Corollary C.2, and from the fact that all the sources $\{X_i\}_{i\in\mathcal{G}}$ are independent. ■

D Computational Network Extractor for polynomially-small Min-Entropy

In this section we give a computational network extractor protocol where each player has an independent (n, k) source with $k = n^{\alpha}$ for some constant $0 < \alpha < 1$. Our protocol works as long as there are a constant number of honest players (the constant depends on α). To describe the protocol, we assume the existence of one way permutations for $0.3n^{\alpha}$-sources. We have the following corollary:

Corollary D.1. *If f is a one way permutation for $0.3n^{\alpha}$-sources, then there is an efficiently computable function* RExt *as in Theorem A.19, with parameters $\epsilon = \frac{1}{\mathrm{poly}(n^{\log n})}$ and $d = O(\log(n/\epsilon)) = \mathrm{polylog}(n)$, such that for any $(n, 0.9n^{\alpha})$ source X,*

$$(\mathsf{RExt}(X, U_d), f(X), U_d) \approx (U_m, f(X), U_d).$$

Proof. The corollary follows from exactly the same proof of Corollary C.2. ■

The computational network extractor protocol is now described as follows:

Protocol D.2. For Computational Network Extractornetwork

Player Inputs: Each player P_i has a string x_i sampled from an independent (n, n^α) source X_i.
Player Outputs: Each player P_i outputs a (private random) string w_i.

Sub-Routines and Parameters:

1. IExt as in Theorem A.21, and $\mathsf{BasicExt}$ as in Theorem A.22.
2. $f : \{0,1\}^n \to \{0,1\}^n$ a one way permutation for $0.3n^\alpha$-sources.
3. The seeded extractor RExt as in Corollary D.1.

The protocol proceeds in two phases.

- **Phase 1.** The first phase of the protocol proceeds in p rounds (where p is the number of players). In round $j \in [p]$ the players do the following.
  1. Player P_j sends $f^{(j+1)}(x_j)$ to all other players. Denote all the j strings broadcasted[a] so far by $y_1, \dots, y_j$. The following steps apply to the remaining players (players P_i with $i > j$).
  2. Let u be the number of independent sources IExt takes. For each $i_1, \dots, i_u \in [j]$, each player P_i computes $m_{i_1,\dots,i_u} \triangleq \mathsf{IExt}(y_{i_1}, \dots, y_{i_u})$. Let M_j be the matrix whose $(i_1, \dots, i_u)$-row is $m_{i_1,\dots,i_u}$. Note that M_j is a (j^u, k)-matrix.
  3. Each player P_i computes $e_i^j = \mathsf{BasicExt}(f^{(j)}(x_i), M_j)$ and truncates the output so that $|e_i^j| = \log^3 n$. Parse $e_i^j = (s_i^j, r_i^j)$.
  4. All players $P_i, i > j$ engage in a secure multi-party computation to compute $r^j = \oplus r_i^j$, where each player P_i uses s_i^j as its internal randomness.
  5. Each player P_i computes $z_i^j = \mathsf{RExt}(f^{(j)}(x_i), r^j)$ and truncates the output so that $|z_i^j| = O(\log^2 n)$.

     At the end of the p'th round, each player $P_i, i \in [p]$ computes $z_i = \oplus_{j=1}^{p} z_i^j$.
- **Phase 2.**
  1. Each player P_i parses $z_i = (s_i, r_i)$. All the players $\{P_i\}_{i\in[p]}$ engage in a secure multi-party computation to compute $r = \oplus r_i$, where each player P_i uses s_i as its internal randomness.
  2. Each player P_i computes $w_i = \mathsf{RExt}(f^{(i)}(x_i), r)$, and outputs w_i as its final output.

[a] For the sake of simplicity, think of the network as having broadcast channels, although our protocol also works in the case of point to point channels.

Theorem D.3. *Let $k = n^\alpha$ for some constant $0 < \alpha < 1$. Let $u = O(\frac{1}{\alpha})$ be the number of independent (n, k) sources* IExt *needs. There exists a constant $0 < \gamma < 1$ such that for any p s.t. $u + 2 \le p \le k^{\gamma/u}$ and any $t \le p - u - 2$, Protocol D.2 is a $(t, p - t)$ computational network extractor.*

To prove the theorem, we first prove the following lemma (in the lemma and the analysis we use capital letters to denote the corresponding strings viewed as random variables):

Lemma D.4. *Let $\ell \in [p]$ be the smallest element such that the set $\{P_1, \ldots, P_\ell\}$ contains at least u honest players: $P_{h_1}, \cdots, P_{h_u}$. Denote the remaining honest players by $P_{g_1}, \cdots, P_{g_v}$. Let e denote the concatenation of all e_i^j's, and b denote the concatenation of the broadcasted sources of all faulty players. Then*

$$(\{Z_{g_i}\}_{i\in[v]}, \{X_{h_i}\}_{i\in[u]}, \{f^{(\ell+1)}(X_{g_i})\}_{i\in[v]}, E, B) \approx (\{U_{g_i}\}_{i\in[v]}, \{X_{h_i}\}_{i\in[u]}, \{f^{(\ell+1)}(X_{g_i})\}_{i\in[v]}, E, B).$$

In other words, at the end of phase 1, the outputs of all the honest players, except the fist u honest players, are indistinguishable from being independent and uniform, even given $X_{h_1}, \cdots, X_{h_u}$, $f^{(\ell+1)}(X_{g_1}), \cdots, f^{(\ell+1)}(X_{g_v})$, all the E_i^j's, and all the sources broadcasted by the faulty players.

Remark D.5. This statement is stronger than the statement that $Z_{g_1}, \cdots, Z_{g_v}$ are indistinguishable from being independent and uniform given all the transcript of Phase 1, because the transcript can be computed in polynomial time from $(X_{h_1}, \cdots, X_{h_u}, f^{(\ell+1)}(X_{g_1}), \cdots, f^{(\ell+1)}(X_{g_v}), E, B)$ (Note that the players $P_{g_1}, \cdots, P_{g_v}$ broadcast their sources after round ℓ, thus $g_i > \ell$. So the broadcasted sources $\{f^{(g_i+1)}(X_{g_i})\}$ can be computed efficiently from $\{f^{(\ell+1)}(X_{g_i})\}$).

Outline of the Proof. We first give an informal outline of the proof, since the proof involves a lot of notations.

We are going to fix the "good" round $\ell = h_u$, where u honest players have broadcasted their sources. We then argue that in this round, the output E_i^ℓ of all honest players that haven't broadcasted their sources are statistically close to uniform, independent of the transcript so far and independent of each other. To do this, note the sources broadcasted by honest players are independent, and each has min-entropy n^α. Thus when we apply IExt to the sources from u honest players, the output will be close to uniform, and therefore, the matrix M_ℓ in round ℓ is close to a somewhere random source. The hope is that when we apply $\mathsf{BasicExt}$ to M_ℓ and a remaining honest player's source, the output will be close to uniform and independent of the transcript so far by Theorem A.22.

However, the transcript contains information(specifically, e_i^j's) about the remaining honest players' sources. Thus we'll have to first fix the transcript, and argue that conditioned on a TYPICAL fixing, a remaining honest player's source and M_ℓ still satisfy the conditions in Theorem A.22. To do this, we first fix $(X_{h_1}, \ldots, X_{h_{u-1}})$. Note that conditioned on this fixing, $\mathsf{IExt}(X_{h_1}, \ldots, X_{h_u})$ is a deterministic function of X_{h_u}. We then show by induction on round $j < \ell$ that conditioned on any fixing of the transcript, X_{h_u} and the remaining honest players' sources are still independent. Moreover, since the size of e_i^j's are small, by Lemma A.12 and Lemma A.13 conditioned a typical fixing of the transcript so far, $\mathsf{IExt}(X_{h_1}, \ldots, X_{h_u})$ is close to a $(k, k - k^\beta)$ source, and any remaining honest player's source is close to an $(n, k - k^\beta)$ source, where β is the constant in Theorem A.22. Thus M_ℓ is close to a $(\ell^u \times k)(k - k^\beta)$-source and is independent of any remaining honest player's source. Now note $\ell < p$, thus as long as p is small, by Theorem A.22 the output E_i^ℓ of all honest players that haven't broadcasted their sources are statistically close to uniform, independent of the transcript so far and independent of each other.

Next, we argue that $Z^{\ell}_{g_i}$ is indistinguishable from being uniform and independent of the transcript so far, and the subsequent computations do not reveal any information about it to a computationally bounded adversary. Thus the final output of any P_{g_i} is indistinguishable from being uniform and private.

To do this, consider a particular honest player P_{g_1}. The fact that there are at least $u+2$ honest players implies there are at least 2 honest players in $\{g_i\}$. Pick another honest player P_{g_2}. Assume for the sake of contradiction that there exists an adversary that distinguishes $Z^{\ell}_{g_1}$ from uniform, given the transcript and the subsequent computations. Then there is a fixing of all the transcript so far(including $E^{\ell}_{g_1}$) and all the sources $\{X_{g_i}\}_{i \neq \{1,2\}}$ such that conditioned on the fixing, the adversary still distinguishes $Z^{\ell}_{g_1}$ from uniform. Note that now all subsequent transcript can be computed in polynomial time from X_{g_2} and $f^{(\ell+1)}(X_{g_1})$. Thus there exists another adversary that distinguishes $Z^{\ell}_{g_1}$ from uniform given X_{g_2} and $f^{(\ell+1)}(X_{g_1})$. Recall $Z^{\ell}_{g_1} = \mathsf{RExt}(f^{(\ell)}(X_{g_1}), R^{\ell})$ and now R^{ℓ} is a deterministic function of X_{g_2}. Thus Lemma A.20 implies there is another adversary that distinguishes $Z^{\ell}_{g_1}$ from uniform given R^{ℓ} and $f^{(\ell+1)}(X_{g_1})$. Since $E^{\ell}_{g_1}$ is statistically close to uniform, the property of the secure multiparty computation guarantees that R^{ℓ} is indistinguishable from being uniform and independent of X_{g_1}. Thus Theorem A.19 implies that there exists an adversary and a weak source $\bar{X}$ with sufficiently large min-entropy such that the adversary inverts $f(\bar{X})$ with non-negligible probability, and this contradicts our assumption on f.

Proof of Lemma D.4. We assume without loss of generality that the PPT adversary (who controls all the malicious players) is deterministic. Thus, it suffice to prove

$$(\{Z_{g_i}\}_{i\in[v]}, \{X_{h_i}\}_{i\in[u]}, \{f^{(\ell+1)}(X_{g_i})\}_{i\in[v]}, E) \approx (\{U_{g_i}\}_{i\in[v]}, \{X_{h_i}\}_{i\in[u]}, \{f^{(\ell+1)}(X_{g_i})\}_{i\in[v]}, E) \qquad (9)$$

Note that after round ℓ, there are u honest players who have already broadcasted their sources. The fact that f is deterministic and injective implies that $Y_{h_1}, \cdots, Y_{h_u}$ are all independent and each has min-entropy $k = n^{\alpha}$. Thus by Theorem A.21, $M_{h_1,\cdots,h_u} = \mathsf{IExt}(Y_{h_1}, \cdots, Y_{h_u})$ is $2^{-k^{\Omega(1)}}$-close to being uniform. We now introduce some notations:

- $E^j = \{E^q_i\}_{i\in[p], q\leq j}$, $Z^j = \{Z^q_i\}_{i\in[p], q\leq j}$, where E^q_i is computed by player P_i in step 3 of round q in phase 1, and Z^q_i is computed by player P_i in step 5 of round q in phase 1. Thus, E^j consists of all E^q_i's computed by all players in all rounds $\leq j$, Z^j consists of all Z^q_i's of all players in all rounds $\leq j$.

Now fix

$$(x_{h_1}, \ldots, x_{h_{u-1}}) \leftarrow (X_{h_1}, \ldots, X_{h_{u-1}})$$

$$(e^{\ell-1}, z^{\ell-1}) \leftarrow (E^{\ell-1}, Z^{\ell-1})$$

For any random variable Z, we denote by Z' the random variable Z conditioned on these fixings. Let TYPICAL denote the event that conditioned on these fixings, the following properties are satisfied:

- $X'_{h_u}, X'_{g_1}, \cdots, X'_{g_v}$ are independent random variables.
- $M'_{h_1,\cdots,h_u}$ is $2^{-k^{\Omega(1)}}$-close to having min-entropy $k - k^{\beta}$.

- $\forall i \in [v]$, X'_{g_i} has min-entropy $k - k^{\beta}$.

Here β is the parameter in Theorem A.22. We have

Claim D.6.

$$\Pr[\text{TYPICAL}] = 1 - \text{negl}(n).$$

The proof of this claim is by induction on $j < \ell$ and is very similar to the proofs of Claim B.3 and Claim B.4, therefore we omit the details here. The only difference is that initially $M_{h_1,\cdots,h_u}$ is only $2^{-k^{\Omega(1)}}$-close to being uniform(having min-entropy k). Thus when dealing with it we need to use Lemma A.13 instead of Lemma A.12.

Now, further fix

$$(x'_{h_u} \leftarrow X'_{h_u})$$

For any random variable Z', we let $Z'' = Z'|(X'_{h_u} = x'_{h_u})$. Let TYPICAL2 denote the event that conditioned on all the above fixings, the following properties are satisfied:

- $\forall i \in [v]$, $(E^{\ell}_{g_i})''$ is $2^{-k^{\Omega(1)}}$-close to being uniform, and is a deterministic function of X''_{g_i}.
- $X''_{g_1}, \cdots, X''_{g_v}$ are independent random variables.

Claim D.7. *If* TYPICAL *holds, then*

$$\Pr[\text{TYPICAL2}] = 1 - \text{negl}(n).$$

As before, the proof of this claim is very similar to the proofs of Claims B.3 and B.4, and therefore we omit the details here. One thing that needs to be noted is that M_{ℓ} is a $\ell^u \times k$ matrix. Thus as long as $p < k^{\gamma/u}$, where γ is the constant in Theorem A.22, the claim follows from Theorem A.22.

Now, assume for the sake of contradiction that Equation (9) does not hold. Namely, assume that there exists a polynomial time non-uniform adversary $\mathcal{A}_1$ and there exists a polynomial q such that for infinitely many n's,

$$\begin{aligned}&|\Pr[\mathcal{A}_1(\{Z_{g_j}\}_{j\in[v]}, \{X_{h_j}\}_{j\in[u]}, \{f^{(\ell+1)}(X_{g_j})\}_{j\in[v]}, E) = 1] - \\ &\Pr[\mathcal{A}_1(\{U_{g_j}\}_{j\in[v]}, \{X_{h_j}\}_{j\in[u]}, \{f^{(\ell+1)}(X_{g_j})\}_{j\in[v]}, E) = 1]| > \frac{1}{q(n)}\end{aligned}$$

A standard hybrid argument implies that there exists $i \in [v]$ such that for infinitely many n's,

$$\begin{aligned}&|\Pr[\mathcal{A}_1(Z_{g_i}, \{Z_{g_j}\}_{j\neq i, j\in[v]}, \{X_{h_j}\}_{j\in[u]}, \{f^{(\ell+1)}(X_{g_j})\}_{j\in[v]}, E) = 1] - \\ &\Pr[\mathcal{A}_1(U, \{Z_{g_j}\}_{j\neq i, j\in[v]}, \{X_{h_j}\}_{j\in[u]}, \{f^{(\ell+1)}(X_{g_j})\}_{j\in[v]}, E) = 1]| > \frac{1}{p \cdot q(n)}\end{aligned}$$

We assume without loss of generality that $i = 1$. Recall that there are at least $u + 2$ honest players. Thus, there are at least 2 honest players in $\{g_j\}_{j\in[v]}$. Pick another honest player P_{g_2}. We say that a tuple

$$(e^{\ell-1}, z^{\ell-1}, e^{\ell}_{g_1}, \{x_{h_j}\}_{j\in[u]}, \{x_{g_j}\}_{j\in[v], j\neq\{1,2\}}) \leftarrow (E^{\ell-1}, Z^{\ell-1}, E^{\ell}_{g_1}, \{X_{h_j}\}_{j\in[u]}, \{X_{g_j}\}_{j\in[v], j\neq\{1,2\}})$$

is BAD if conditioned on the fixing of this tuple, the following properties are satisfied:

1. There exists a non-uniform polynomial time adversary $\mathcal{A}_2$ such that for infinitely many n's,

$$\left|\Pr[\mathcal{A}_2(Z_{g_1}^\ell, R^\ell, X_{g_2}, f^{(\ell+1)}(X_{g_1})) = 1] - \Pr[\mathcal{A}_2(U, R^\ell, X_{g_2}, f^{(\ell+1)}(X_{g_1})) = 1]\right| \geq \frac{1}{2p \cdot q(n)},$$

2. X_{g_1} and X_{g_2} are independent. $E_{g_2}^\ell$ is a deterministic function of X_{g_2} and is $2^{-k^{\Omega(1)}}$-close to being uniform. X_{g_1} has min-entropy $k - o(k)$.

Claim D.8. *There exists a BAD tuple.*

Again, the proof of this claim is rather standard and is very similar to the proof of Claim C.4, we thus omit the details here.

Now fix a BAD tuple $(e^{\ell-1}, z^{\ell-1}, e_{g_1}^\ell, \{x_{h_j}\}_{j\in[u]}, \{x_{g_j}\}_{j\in[v],j\neq\{1,2\}})$. Then all R_j^ℓ's from honest players except P_{g_2} are fixed. The R_j^ℓ's from faulty players are a deterministic function of the transcript so far, thus are also fixed. Note $R^\ell = \bigoplus R_j^\ell$ and $R_{g_2}^\ell$ is a deterministic function of X_{g_2}. Thus R^ℓ is now a deterministic function of X_{g_2}. Note X_{g_1} and X_{g_2} are independent conditioned on the fixing, thus R^l is independent of X_{g_1}. Moreover, since $E_{g_2}^\ell$ is $2^{-k^{\Omega(1)}}$-close to being uniform, the property of the secure multiparty computation protocol guarantees that the R_j^ℓ's from faulty players are indistinguishable from being independent of $R_{g_2}^\ell$. Thus $R^\ell = \bigoplus R_i^\ell$ is indistinguishable from being uniform. Note $Z_{g_1}^\ell = \mathsf{RExt}(f^{(\ell)}(X_{g_1}), R^\ell)$, thus

$$(Z_{g_1}^\ell, R^\ell, f^{(\ell+1)}(X_{g_1})) \approx (\mathsf{RExt}(f^{(\ell)}(X_{g_1}), R), R, f^{(\ell+1)}(X_{g_1})), \tag{10}$$

where R is the uniform distribution on the range of R^ℓ and is independent of X_{g_1}.

On the other hand, note that when we fix the BAD tuple, X_{g_1} and X_{g_2} are independent, and R^ℓ is a deterministic function of X_{g_2}. Thus by the first property of the BAD tuple and Lemma A.20, there exists another non-uniform adversary $\mathcal{A}_3$ that runs in time $2^{|R^\ell|} n Time(\mathcal{A}_2) = \mathrm{poly}(n, \frac{1}{\epsilon}) Time(\mathcal{A}_2)$ such that

$$\left|\Pr[\mathcal{A}_3(Z_{g_1}^\ell, R^\ell, f^{(\ell+1)}(X_{g_1})) = 1] - \Pr[\mathcal{A}_3(U, R^\ell, f^{(\ell+1)}(X_{g_1})) = 1]\right| \geq \frac{1}{2p \cdot q(n)}.$$

Note $R_l \approx R$, combined with Equation 10 we get

$$\left|\Pr[\mathcal{A}_3(\mathsf{RExt}(f^{(\ell)}(X_{g_1}), R), R, f^{(\ell+1)}(X_{g_1})) = 1] - \Pr[\mathcal{A}_3(U, R, f^{(\ell+1)}(X_{g_1})) = 1]\right| \geq \frac{1}{2p \cdot q(n)} - \mathrm{negl(n)} > \frac{1}{3\mathrm{p} \cdot \mathrm{q(n)}}.$$

Note $\frac{1}{3p \cdot q(n)} > 2\epsilon$ and $f^{(\ell)}(X_{g_1})$ has min-entropy $k - o(k) > 0.9k = 0.9n^\alpha$ conditioned on all the fixings. Therefore, by Theorem A.19 there exists a non-uniform adversary $\mathcal{A}_4$ that runs in time $\mathrm{poly}(n, 1/\epsilon) \cdot Time(\mathcal{A}_3) = \mathrm{poly}(n, 1/\epsilon) Time(\mathcal{A}_2) = \mathrm{poly}(n^{\log n})$ and an $(n, 0.3n^\alpha)$-source $\bar{X}$ such that $\mathcal{A}_4$ inverts $f(\bar{X})$ with probability at least $\frac{1}{24p \cdot q(n)}$. This contradicts our assumption on f. ■

Once we have the lemma, it's fairly easy to prove the main theorem. We first prove that if $\{Z_{g_i}\}$'s are really $\{U_{g_i}\}$'s, then all W_i's of honest players are indistinguishable from being uniform and private. Then since $\{Z_{g_i}\}$'s are indistinguishable from $\{U_{g_i}\}$'s, the theorem follows.

To prove the statement above, consider any particular honest player P_j. Assume there exists a PPT adversary that distinguishes W_j and uniform given the transcript in Phase 1 and Phase 2. We first fix all honest players' sources except P_j. There is a fixing of the sources such that the adversary still distinguishes W_j and uniform given the transcript. Note after this fixing all transcript in Phase 1 and Phase 2 except $\{U_{g_i}\}$'s are a deterministic function of X_j. Now we further fix all the transcript and $\{U_{g_i}\}$'s except $f^{(j+1)}(X_j)$ and U_{g_1}. Again there is a fixing such that the adversary still distinguishes W_j and uniform. Now the adversary is only given $f^{(j+1)}(X_j)$ and U_{g_1}. Recall $W_j = \mathsf{RExt}(f^{(j)}(X_j), R)$ and note now R is a deterministic function of U_{g_1}. Thus Lemma A.20 implies there exists another adversary that distinguishes W_j and uniform given R and $f^{(j+1)}(X_j)$. The property of the secure multiparty computation guarantees that R is indistinguishable from being uniform and independent of X_j. Note all E_i^j's and Z_i^j's are small thus conditioned on all the fixings mentioned above $f^{(j)}(X_j)$ still has min-entropy $> 0.9k$. Thus Theorem A.19 implies that there exists another adversary and a weak source $\bar{X}$ with sufficiently large min-entropy such that the adversary inverts $f(\bar{X})$ with non-negligible probability. This contradicts our assumption on f.

Proof of Theorem D.3. Again, we assume without loss of generality that the PPT adversary (who controls all the malicious players) is deterministic. At the end of Phase 1, we have

$$(\{Z_{g_i}\}_{i\in[v]}, \{X_{h_i}\}_{i\in[u]}, \{f^{(\ell+1)}(X_{g_i})\}_{i\in[v]}, E) \approx (\{U_{g_i}\}_{i\in[v]}, \{X_{h_i}\}_{i\in[u]}, \{f^{(\ell+1)}(X_{g_i})\}_{i\in[v]}, E)$$

Let the set of all honest players be $\mathcal{G}$, i.e., $\mathcal{G} = \{h_i\} \cup \{g_i\}$. Note that $(\{Z_{h_i}\}_{i\in[u]}, \{f^{(h_i)}(X_{h_i})\}_{i\in[u]}, \{f^{(g_i)}(X_{g_i})\}_{i\in[v]})$ can be computed in polynomial time from $(\{X_{h_i}\}_{i\in[u]}, \{f^{(\ell+1)}(X_{g_i})\}_{i\in[v]}, E)$ (keep in mind that $g_i \geq l+1$). Thus we have

$$(\{Z_{g_i}\}_{i\in[v]}, \{Z_{h_i}\}_{i\in[u]}, \{f^{(i)}(X_i)\}_{i\in\mathcal{G}}, E) \approx (\{U_{g_i}\}_{i\in[v]}, \{Z_{h_i}\}_{i\in[u]}, \{f^{(i)}(X_i)\}_{i\in\mathcal{G}}, E) \tag{11}$$

Note that the transcript in Phase 1 can be computed in polynomial time from $(E, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}})$, and the transcript in Phase 2 can be computed in polynomial time from $(\{Z_i\}_{i\in[p]})$. Moreover, $(\{Z_i\}_{i\notin\mathcal{G}})$ can be computed in polynomial time from the transcript in Phase 1. Thus to prove the theorem it suffices to prove

$$\begin{aligned}&(\{W_i\}_{i\in\mathcal{G}}, \{Z_i\}_{i\in\mathcal{G}}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) \approx\\ &(\{U_i\}_{i\in\mathcal{G}}, \{Z_i\}_{i\in\mathcal{G}}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E).\end{aligned}$$

Now if we run Phase 2 with the two distributions on both sides of Equation 11, and let $\bar{w}_i$ denote the output of player P_i when we run the protocol with the right hand side distribution, we get

$$\begin{aligned}&(\{W_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{Z_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) \approx\\ &(\{\bar{W}_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E).\end{aligned} \tag{12}$$

We'll first prove

$$(\{\bar{W}_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) \approx$$
$$(\{U_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E).$$

Assume for the sake of contradiction that there exists a non-uniform polynomial time adversary $\mathcal{A}_1$ and a polynomial q such that for infinitely many n's,

$$|\Pr[\mathcal{A}_1(\{\bar{W}_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) = 1] -$$
$$\Pr[\mathcal{A}_1(U_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) = 1]| > \frac{1}{q(n)}$$

A standard hybrid argument implies that there exists $j \in \mathcal{G}$ such that

$$|\Pr[\mathcal{A}_1(\bar{W}_j, \{\bar{W}_i\}_{i\in\mathcal{G}, i\neq j}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) = 1] -$$
$$\Pr[\mathcal{A}_1(U, \{\bar{W}_i\}_{i\in\mathcal{G}, i\neq j}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) = 1]| > \frac{1}{p\cdot q(n)}$$

for infinitely many n's.
We say that a tuple

$$(e, \{z_{h_i}\}_{i\in[u]}, \{x_i\}_{i\neq j, i\in\mathcal{G}}) \leftarrow (E, \{Z_{h_i}\}_{i\in[u]}, \{X_i\}_{i\neq j, i\in\mathcal{G}})$$

is BAD if conditioned on the fixing of this tuple, the following properties are satisfied:

1. There exits a non-uniform polynomial time adversary $\mathcal{A}_2$ such that for infinitely many n's

$$\left|\Pr[\mathcal{A}_2(\bar{W}_j, \{U_{g_i}\}_{i\in[v]}, f^{(j+1)}(X_j)) = 1] - \Pr[\mathcal{A}_2(U, \{U_{g_i}\}_{i\in[v]}, f^{(j+1)}(X_j)) = 1]\right| \geq \frac{1}{2p\cdot q(n)},$$

2. X_j has min-entropy $k - o(k)$.

Claim D.9. *There exists a BAD tuple.*

Proof of Claim D.9. A standard probabilistic argument shows that a random tuple

$$(e, \{z_{h_i}\}_{i\in[u]}, \{x_i\}_{i\neq j, i\in\mathcal{G}}) \leftarrow (E, \{Z_{h_i}\}_{i\in[u]}, \{X_i\}_{i\neq j, i\in\mathcal{G}})$$

satisfies

$$|\Pr[\mathcal{A}_1(\bar{W}_j, \{\bar{W}_i\}_{i\in\mathcal{G}, i\neq j}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) = 1] -$$
$$\Pr[\mathcal{A}_1(U, \{\bar{W}_i\}_{i\in\mathcal{G}, i\neq j}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) = 1]| \geq \frac{1}{2p\cdot q(n)}$$

with probability at least $\frac{1}{2p\cdot q(n)}$.

Note that once $(E, \{Z_{h_i}\}_{i\in[u]}, \{X_i\}_{i\neq j, i\in\mathcal{G}})$ are fixed, $(\{\bar{W}_i\}_{i\in\mathcal{G}, i\neq j}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}, i\neq j})$ can be computed in polynomial time from $\{U_{g_i}\}_{i\in[v]}$. Thus there exists a non-uniform adversary PPT $\mathcal{A}_2$ that has the fixings hardwired into it such that

$$\left|\Pr[\mathcal{A}_2(\bar{W}_j, \{U_{g_i}\}_{i\in[v]}, f^{(j+1)}(X_j)) = 1] - \Pr[\mathcal{A}_2(U, \{U_{g_i}\}_{i\in[v]}, f^{(j+1)}(X_j)) = 1]\right| \geq \frac{1}{2p\cdot q(n)}$$

Furthermore, since $\{X_i\}_{i\in\mathcal{G}}$ are independent, it's easy to show by induction on round $j'\in[p]$ that the only fixings that can cause X_j to lose entropy are $E_j^{j'}$'s and Z_j, and these are a deterministic function of X_j(conditioned on the fixings). The total length of these strings is at most $O(p\log^3 n) = o(k)$ since $k = n^\alpha$ and $p \leq k^{\gamma/u}$. Thus by Lemma A.12 with probability $1-\mathrm{negl}(n)$ over the fixings of $(E, \{Z_{h_i}\}_{i\in[u]}, \{X_i\}_{i\neq j, i\in\mathcal{G}})$, X_j has min-entropy $k - o(k)$. The claim thus follows. ■

Now we further fix $\{U_{g_i}\}_{i\in[v]}$ except U_{g_1}. There is a fixing of $\{U_{g_i}\}_{i\in[v], i\neq 1}$ that preserves this probability. Thus there exists a non-uniform PPT adversary $\mathcal{A}_3$ that has the fixings hardwired into it such that conditioned on the fixings,

$$\left|\Pr[\mathcal{A}_3(\bar{W}_j, U_{g_1}, f^{(j+1)}(X_j)) = 1] - \Pr[\mathcal{A}_3(U, U_{g_1}, f^{(j+1)}(X_j)) = 1]\right| \geq \frac{1}{2p\cdot q(n)}$$

for infinitely many n's.

Moreover, after all these fixings R is a deterministic function of U_{g_1}. Note that $\bar{W}_j = \mathsf{RExt}(f^{(j)}(X_j), R)$ and U_{g_1} is independent of X_j. Thus by Lemma A.20 there exists a non-uniform adversary $\mathcal{A}_4$ that runs in time $2^{|R|}nTime(A_3) = \mathrm{poly}(n, \frac{1}{\epsilon})Time(A_3) = \mathrm{poly}(n^{\log n})$ such that

$$\left|\Pr[\mathcal{A}_4(\bar{W}_j, R, f^{(j+1)}(X_j)) = 1] - \Pr[\mathcal{A}_4(U, R, f^{(j+1)}(X_j)) = 1]\right| \geq \frac{1}{2p\cdot q(n)}.$$

Note $\frac{1}{2p\cdot q(n)} > 2\epsilon$. Since U_{g_1} is uniform and independent of all the other random variables, the property of the secure multiparty computation protocol guarantees that $R = \bigoplus R_i$ is indistinguishable from being uniform and independent of X_j. Further note conditioned on all the fixings above, $f^{(j)}(X_j)$ has min-entropy $k - o(k) > 0.9k$. Thus by Theorem A.19 there exists a non-uniform adversary $\mathcal{A}_5$ that runs in time $\mathrm{poly}(n, \frac{1}{\epsilon})Time(A_4) = \mathrm{poly}(n^{\log n})$ and an $(n, 0.3n^\alpha)$-source $\bar{X}$ such that $\mathcal{A}_5$ inverts $f(\bar{X})$ with probability at least $\frac{1}{16p\cdot q(n)}$. This contradicts our assumption on f.

Therefore, we must have

$$\begin{aligned}&(\{\bar{W}_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) \approx\\ &(\{U_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E).\end{aligned}$$

From Equation 11 we get

$$\begin{aligned}&(\{U_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) \approx\\ &(\{U_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{Z_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E).\end{aligned}$$

Therefore

$$(\{\bar{W}_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{U_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) \approx$$
$$(\{U_i\}_{i\in\mathcal{G}}, \{Z_{h_i}\}_{i\in[u]}, \{Z_{g_i}\}_{i\in[v]}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E).$$

Together with Equation 12 this implies

$$(\{W_i\}_{i\in\mathcal{G}}, \{Z_i\}_{i\in\mathcal{G}}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E) \approx$$
$$(\{U_i\}_{i\in\mathcal{G}}, \{Z_i\}_{i\in\mathcal{G}}, \{f^{(i+1)}(X_i)\}_{i\in\mathcal{G}}, E).$$

as desired. ■

E Proof of Lemma A.11

In this section, we prove Lemma A.11, which says that any weak source with linear min-entropy can be divided into a constant number of blocks, such that the source is close to a convex combination of somewhere block sources. First we need the definition of a subsource.

Definition E.1 (Subsource)**.** Given random variables X and X' on $\{0,1\}^n$ we say that X' is a *deficiency-d subsource* of X and write $X' \subseteq X$ if there exits a set $A \subseteq \{0,1\}^n$ such that $(X|A) = X'$ and $\Pr[x \in A] \geq 2^{-d}$.

Proposition E.2. *Let X be a random variable with $H_\infty(X) = k$. Let $X' \subset X$ be a subsource of deficiency d corresponding to some set $A \subset \{0,1\}^n$. Then $H_\infty(X') = k - d$.*

More generally, we have the statement that conditioning on *typical* values of any function cannot reduce the min-entropy of our source by much more than we expect.

Lemma E.3 (Fixing a function)**.** *Let X be a distribution over $\{0,1\}^n$, $F : \{0,1\}^n \to \{0,1\}^m$ be a function, and $\ell \geq 0$ some number. For every $s \in \mathsf{supp}(F(X))$, define X_s to be the subsource $X|F(X) = s$. Then there exists $s \in \{0,1\}^m$ for which X_s has deficiency at most m. Furthermore, we have that*

$$\Pr_{s\leftarrow_R F(X)}[\textit{deficiency of } X_s \leq m + \ell] \geq 1 - 2^{-\ell}$$

Proof. Let S be the set of $s \in \{0,1\}^m$ such that $\Pr[F(x) = s] < 2^{-m-\ell}$. Since $|S| \leq 2^m$, we have that $\Pr[F(X) \in S] < 2^{-\ell}$. If we choose $s \leftarrow_{\mathrm{R}} F(X)$ and $s \notin S$, we get that $X|F(X) = s$ has deficiency $\leq m + \ell$. Choosing $\ell = 0$ we get the first part of the proposition. ■

We next give a lemma that is used to prove Lemma A.11.

Lemma E.4 (Fixing Entropies)**.** *Let $X = X_1 \circ X_2 \circ \cdots \circ X_t$ be a t-block random variable over $\{0,1\}^n$. Fix any $s > 0$ and let $0 = \tau_1 < \tau_2 < \cdots < \tau_{c+1} = n$ be some numbers. There exists a universe U such that for every X there exists a set of random variables $\{X^j\}_{j\in U}$ and a random variable J over U, such that $X = X^J$ (i.e., X is a convex combination of $\{X^j\}_{j\in U}$). $\{X^j\}$ has the following properties:*

- *For every $j \in U$ s.t.* $\Pr[J = j] > 0$, *there exists a sequence* $\bar{e^j} = e^j_1, \cdots, e^j_t \in [c]^t$ *such that for every* $0 < i \le t$ *and every sequence* $x_1, \cdots, x_{i-1} \in \mathsf{Supp}(X^j_{1,\cdots,i-1})$*;*

$$\tau_{e^j_i} < H_\infty(X^j_i | x_1, \cdots, x_{i-1}) \le \tau_{e^j_i+1}$$

- *with probability* $1 - t2^{-s}$ *over* J, X^j *is a subsource of* X *with deficiency* $< t^2 \log c + ts$*.*

Proof. We prove this by induction on t. The base case where $t = 1$ is trivially true. Now suppose this is true for up to $t-1$ blocks and we'll prove it for t blocks. For every $x_1 \in \mathsf{Supp}(X_1)$ define the source $Y(x_1)$ to be $X_{2,\cdots,t}|x_1$. By the induction hypothesis, there exists a universe U' and a random variable J' over U' such that $Y(x_1) = Y^{J'}$. For every $j' \in U'$ s.t. $\Pr[J' = j'] > 0$ there exists a sequence $\bar{e^{j}}'(x_1) \in [c]^{t-1}$ such that $Y^{j'}$ satisfies the first property with respect to $\bar{e^{j}}'(x_1)$. Define the function $F_{j'} : X_1 \to [c]^{t-1}$ that maps x_1 to $\bar{e^{j}}'(x_1)$.

Now let the new universe be $U = \mathbf{Range}(F(X_1)) \times U'$. Note that U is the same for all X. Define the new random variable J over U such that the event $J = (\bar{e}, j')$ stands for $(J' = j', F_{j'}(X_1) = \bar{e})$. Then the convex combination $X = X^J$ satisfies property 1. Moverover, by Lemma E.3, with probability $1 - 2^{-s}$, $X_1 | F_{j'}(X_1) = \bar{e}$ is a deficiency $(t-1)\log c + s$ subsource of X_1, and by the induction hypothesis with probability $1-(t-1)2^{-s}$ over J', $Y^{j'}$ is a deficiency $(t-1)^2 \log c + (t-1)s$ subsource of $Y(x_1)$. Thus with probability at least $1 - (t-1)2^{-s} - 2^{-s} = 1 - t2^{-s}$, the deficiency of X^j is at most $(t-1)^2 \log c + (t-1)s + (t-1)\log c + s < t^2 \log c + ts$. ■

Corollary E.5. *If in the lemma above* X *has min-entropy* k*, and* X^j *is a deficiency* $t^2 \log c + ts$ *subsource of* X *as in property 2 with* $\bar{e^j}$ *the sequence corresponding to* X^j *as in property 1, then* $\sum_{i=1}^t \tau_{e^j_i+1} \ge k - t^2 \log c - ts$*.*

Proof. If this was the case, we could find some string in the support of X that is too heavy. Specifically, we take the heaviest string allowed in each successive block to get $x = x_1 \circ x_2 \circ \cdots \circ x_t$. Then it must be $\Pr[X_i = x_i | x_1, \cdots, x_{i-1}] \ge 2^{-\tau_{e^j_i+1}}$ for any $0 < i \le t$. Together with the fact that X^j has deficiency $< t^2 \log c + ts$ we get $\Pr[X = x] > 2^{-(t^2 \log c + ts)} \Pi_{i=1}^t 2^{-\tau_{e^j_i+1}} = 2^{-(t^2 \log c + ts)} 2^{-\sum_{i=1}^t \tau_{e^j_i+1}} > 2^{-k}$. This contradicts the fact that X has min-entropy k. ■

Proof of Lemma A.11. We'll use Lemma E.4. Let the parameters in that lemma be $s = \sqrt{k}$, $c = \frac{6}{\alpha^2}$ and $\tau_i = \frac{i-1}{c} n$ for $0 < i \le c+1$. Then Lemma E.4 shows that X is a convex combination of $\{X^j\}_{j \in U}$ and with probability $1 - t2^{-s} = 1 - 2^{-n^{\Omega(1)}}$, X_j is a subsource with deficiency $< t^2 \log c + ts < 0.01k$. Now Corollary E.5 says that for such a X^j, we must have $\sum_{i=1}^t \tau_{e^j_i+1} \ge k - t^2 \log c - ts > 0.99k$. We now show that there must exist at least two e^j_i's s.t. $e^j_i \ge 2$. Otherwise suppose there is at most one e^j_i s.t. $e^j_i \ge 2$. For $e^j_i = 1$ we have $\tau_{e^j_i+1} = \tau_2 = \frac{n}{c}$. For $e^j_i \ge 2$ we have the min-entropy of the block X^j_i conditioned on any fixing of previous blocks is at most $\frac{n}{t}$. Assume for the sake of simplicity that $\frac{n}{ct} = \frac{1.5}{\alpha}$ is an integer, thus $\frac{n}{t}$ appears in set $\{\tau_i\}$ and we must have $\tau_{e^j_i+1} \le \frac{n}{t}$. Therefore $\sum_{i=1}^t \tau_{e^j_i+1} \le \frac{n}{c}(t-1) + \frac{n}{t} < \frac{tn}{c} + \frac{n}{t} = \frac{2\alpha}{3} n + \frac{\alpha}{4} n < 0.99\alpha n = 0.99k$, which is a contradiction.

Thus, there must exist at least two e^j_i's s.t. $e^j_i \ge 2$, so $\tau_{e^j_i} \ge \frac{n}{c} = \frac{\alpha^2}{6} n$. Let $0 < i_1 < i_2 \le t$ be the two corresponding i's. Let $g = i_1$ and further condition on any fixing of $X^j_1, \ldots, X^j_{g-1}$. Now by Lemma E.4, we see X is $2^{-\Omega(n)}$-close to being a convex combination of sources $\{X^j\}_{j\in J}$ that satisfy the properties in Lemma A.11. ■

F Proof of Theorem A.19

Following [Uma05], we define reconstructive extractors:

Definition F.1. A $(n, t, m, d, a, \epsilon, \delta)$-reconstructive extractor is a triple of functions:

- A polynomial time computable extractor function $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$
- An advice function $\mathsf{A} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^a$
- A $\mathrm{poly}(n, 1/\epsilon)$ time randomized oracle reconstruction procedure $\mathcal{R} : \{0,1\}^a \to \{0,1\}^n$

That satisfy the property that for every $x \in \{0,1\}^n$ and $\mathcal{D} : \{0,1\}^m \to \{0,1\}$ for which

$$|\Pr[\mathcal{D}(\mathsf{Ext}(x, U_t), U_t) = 1] - \Pr[\mathcal{D}(U_m, U_t) = 1]| \geq \epsilon,$$

we must have that

$$\Pr_w[\mathcal{R}^{\mathcal{D}}(A(x, w)) = x] \geq \delta.$$

Note that Ext as above must be a seeded extractor for n bit sources with entropy larger than a, since any function that distinguishes the output from uniform can be used to get a procedure that guesses x with probability roughly 2^{-a}.

We have the following theorem, which follows from the discussion in section 6 of [Uma05]:

Theorem F.2 ([Uma05])**.** *There is a constant $\beta > 0$ such that for every n, a, ϵ with $a = n^{\Omega(1)}$, there exists $(n, t = O(\log(n/\epsilon)), m = n^\beta, d = O(\log(n/\epsilon)), a, \epsilon, 1/2)$ reconstructive extractor.*

An almost immediate consequence of this theorem is Theorem A.19, which we prove here:

Proof of Theorem A.19. We set Ext to be the reconstructive extractor promised by Theorem F.2, set up so that $a = k/2$. Suppose that there was a distinguishing circuit D that could distinguish $(f(X), \mathsf{Ext}(X, U_t), U_t)$ from $(f(X), U_m, U_t)$ with probability $2\epsilon_1$. Then by a standard averaging argument we have

$$\Pr_{x \leftarrow X}[|\Pr[\mathcal{D}(f(x), \mathsf{Ext}(x, U_t), U_t) = 1] - \Pr[\mathcal{D}(f(x), U_m, U_t) = 1]| \geq \epsilon_1] \geq \epsilon_1$$

Note $\epsilon_1 \geq \epsilon$. Thus by the definition, there is a circuit R^D of size $\mathrm{poly}(n, 1/\epsilon)\mathsf{size}(D)$ such that for every x s.t. $|\Pr[\mathcal{D}(f(x), \mathsf{Ext}(x, U_t), U_t) = 1] - \Pr[\mathcal{D}(f(x), U_m, U_t) = 1]| \geq \epsilon_1 \geq \epsilon$, $\Pr[R^D(f(x), A(x, W)) = x] = 1/2$. Thus we have

$$\Pr[R^D(f(X), A(X, W)) = X] \geq \frac{\epsilon_1}{2},$$

where the probability is over X and W. By Lemma A.12,

$$\Pr_{a \leftarrow_{\mathrm{R}} A(X,W)}[H_\infty(X|A(X, W) = a) \geq k/3] \geq 1 - 2^{a+k/3-k} = 1 - 2^{-\Omega(k)}.$$

Also, by averaging, we have that

$$\Pr_{a \leftarrow_{\mathrm{R}} A(X,W)}[\Pr[R^D(f(X), A(X,W)) = X | A(X,W) = a] \geq \frac{\epsilon_1}{4}] \geq \frac{\epsilon_1}{4}.$$

Note $\epsilon_1 \geq \epsilon \geq 2^{-\sqrt{k}}$. Thus, by a union bound, there is some fixing of $A(X,W) = a$ for which $H_\infty(X|A(X,W) = a) \geq k/3$ and $\Pr[R^D(f(X), A(X,W)) = X|A(X,W) = a] \geq \epsilon_1/4$. Let $\bar{X} = X|(A(X,W) = a)$ and $\mathcal{B}$ be R^D with $A(X,W) = a$ hardwired into it. The theorem now follows. ■