2015 IEEE 56th Annual Symposium on Foundations of Computer Science

Three-Source Extractors for Polylogarithmic Min-Entropy

Xin Li
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21218, U.S.A.
lixints@cs.jhu.edu

Abstract

We continue the study of constructing explicit extractors for independent general weak random sources. The ultimate goal is to give a construction that matches what is given by the probabilistic method — an extractor for two independent n-bit weak random sources with min-entropy as small as $\log n + O(1)$. Previously, the best known result in the two-source case is an extractor by Bourgain [1], which works for min-entropy $0.49n$; and the best known result in the general case is an earlier work of the author [2], which gives an extractor for a constant number of independent sources with min-entropy $\mathrm{polylog}(n)$. However, the constant in the construction of [2] depends on the hidden constant in the best known seeded extractor, and can be large; moreover the error in that construction is only $1/\mathrm{poly}(n)$.

In this paper, we make two important improvements over the result in [2]. First, we construct an explicit extractor for *three* independent sources on n bits with min-entropy $k \geq \mathrm{polylog}(n)$. In fact, our extractor works for one source with poly-logarithmic min-entropy and another independent block source with two blocks each having poly-logarithmic min-entropy. This significantly improves previous constructions, and the next step would be to break the $0.49n$ barrier in two-source extractors. Second, we improve the error of the extractor from $1/\mathrm{poly}(n)$ to $2^{-k^{\Omega(1)}}$, which is almost optimal and crucial for cryptographic applications. Some of our techniques may be of independent interests.

Keywords

extractor; randomness; independent source

I. Introduction

Randomness extractors are fundamental objects in studying the role of randomness in computation. Motivated by the wide applications of randomness in computation, the standard requirements that the randomness used should be uniform, and the fact that real world random sources are almost always biased and defective, randomness extractors are functions that transform imperfect random sources into nearly uniform random bits. In addition, these objects are especially useful in cryptographic applications, since there even originally uniform random secrets can be compromised as a result of side channel attacks. To formally define randomness extractors, we model imperfect randomness as an arbitrary probability distribution with a certain amount of entropy.

Definition I.1. The *min-entropy* of a random variable X is

$$H_\infty(X) = \min_{x \in \mathsf{supp}(X)} \log_2(1/\Pr[X = x]).$$

For $X \in \{0,1\}^n$, we call X an $(n, H_\infty(X))$-source, and we say X has *entropy rate* $H_\infty(X)/n$.

It is easy to show that no deterministic extractor can work for all (n,k) sources even when $k = n-1$. Thus the study of randomness extractors has taken two different approaches. The first is to give the extractor an additional independent uniform random seed. These extractors are called *seeded extractors* and were introduced by Nisan and Zuckerman [3].

Definition I.2. (Seeded Extractor) A function $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is a (k,ε)-*extractor* if for every source X with min-entropy k and independent Y which is uniform on $\{0,1\}^d$,

$$|\mathsf{Ext}(X,Y) - U_m| \leq \epsilon.$$

If in addition we have $|(\mathsf{Ext}(X,Y),Y) - (U_m,Y)| \leq \epsilon$ then we say it is a *strong* (k,ε)-*extractor*.

One can show that with a very short random seed (typically of length say $d = O(\log n)$), it is possible to construct extractors for all weak random sources. Moreover, even without the auxiliary random seed, these extractors can be

0272-5428/15 $31.00 © 2015 IEEE
DOI 10.1109/FOCS.2015.58

used in many applications (such as simulating randomized algorithms using weak random sources) just by trying all possible values of the seed. Seeded extractors have also been found to be related to many other areas in computer science, and today we have nearly optimal constructions of such extractors (e.g., [4]–[7]).

However, seeded extractors are not enough for many other important applications, most notably the ones in distributed computing and cryptography, where the trick of trying all possible values of the seed does not work. Instead, in these applications we need extractors without the uniform random seed. These extractors are called *seedless extractors*. Given that it is impossible to build extractors that use just a single weak random source, one natural alternative is to try to build extractors that use multiple independent weak random sources. Indeed, it seems reasonable to assume that we can find more than one independent weak sources in nature, such as stock market, thermal noise, computer mouse movements and so on. Such extractors are called independent source extractors. A formal definition is given below.

Definition I.3 (Independent Source Extractor)**.** A function $\mathsf{IExt} : (\{0,1\}^n)^t \rightarrow \{0,1\}^m$ is an extractor for independent (n,k) sources that uses t sources and outputs m bits with error ϵ, if for any t independent (n,k) sources $X_1, X_2, \cdots, X_t$, we have

$$|\mathsf{IExt}(X_1, X_2, \cdots, X_t) - U_m| \leq \epsilon.$$

Constructing independent source extractors is a major problem in the area of *pseudorandomness*, and has been studied for a long time. Indeed these extractors have been used in distributed computing and cryptography (e.g., the network extractor protocols in [8], [9]). Here, one natural goal is to construct extractors that use as few number of sources as possible. For example, in [10], Chor and Goldreich showed that the well known Lindsey's lemma gives an extractor for two independent (n,k) sources with $k > n/2$. One can also use the probabilistic method to show that there exists a deterministic extractor for just two independent sources with logarithmic min-entropy, which is optimal since extractors for one weak source do not exist. In fact, the probabilistic method shows that with high probability a random function is such a two-source extractor. Thus, explicit constructions of independent source extractors is also closely related to the general problem of *derandomization*.

Independent source extractors also have close connections to Ramsey graphs. For example, given any boolean function with two n-bit inputs, one can construct a bipartite graph with $N = 2^n$ vertices on each side, such that two vertices are connected if and only if the output is 1. If the function is a two-source extractor for (n,k) sources, then the resulted bipartite graph has no bipartite clique or independent set of size $K = 2^k$ (i.e., a Ramsey graph). With some extra efforts, this bipartite Ramsey graph can also be converted to a regular Ramsey graph. Before our work, the best construction is due to Barak et. al [11], which gives a Ramsey graph that has no clique or independent set of size $2^{2^{(\log\log n)^{1-\alpha}}}$ for some constant $\alpha > 0$. More generally, extractors that use a few (say a constant) number of sources give Ramsey hypergraphs.

Finally, independent source extractors are also quite useful in constructing seedless extractors for other structured sources, because in many cases other structured sources can be reduced to independent sources. Two such examples are the constructions of extractors for affine sources in [12] and extractors for small space sources in [13].

However, despite considerable efforts spent on this problem, the known constructions of two-source extractors are far from optimal. To date the best known two-source extractor due to Bourgain [1], only works for entropy $k \geq (1/2 - \delta)n$ for some small universal constant $\delta > 0$. Quantitatively, this only slightly improves the result of Chor and Goldreich [10]. Given this difficult situation, researchers have turned to the alternative approach of constructing extractors that use a few more weak random sources, and ideally ones that only use a constant number of sources.

This approach has been quite fruitful, starting from the work of Barak, Impagliazzo and Wigderson [14], who showed how to extract from a constant number $(\mathrm{poly}(1/\delta))$ of independent $(n, \delta n)$ sources, for any constant $\delta > 0$. Following this work, Barak et al. [15] constructed extractors for three independent $(n, \delta n)$ sources for any constant $\delta > 0$. This was later improved by Raz [16] to given an extractor that works for three independent sources where only one is required to be an $(n, \delta n)$ source while the other two can have entropy as small as $k \geq \mathrm{polylog}(n)$. In the same paper Raz also gave an extractor for two independent sources where one is required to have entropy $k \geq (1/2 + \delta)n$ for any constant $\delta > 0$, and the other can have entropy as small as $k \geq \mathrm{polylog}(n)$. Most of these work use advanced techniques in additive combinatorics, such as sum-product theorems and incidence theorems.

However, these results only achieve a constant number of sources if at least one source has min-entropy δn for any constant $\delta > 0$.

Using clever ideas related to somewhere random sources, Rao [17] and subsequently Barak et al. [11] constructed extractors for (n, k) sources that use $O(\log n / \log k)$ independent sources. In particular, these extractors only use a constant number of sources even if the min-entropy is n^δ for any constant $\delta > 0$. Based on these techniques, in [18] the author gave an extractor for three independent (n, k) sources with $k \geq n^{1/2+\delta}$ for any constant $\delta > 0$. However, in the worst case where $k = \mathrm{polylog}(n)$, the number of sources required is still super-constant (i.e., $O(\log n / \log \log n)$).

In [2], [19], the author further exploited the properties of somewhere random sources and established a connection between extraction from such sources and the problem of leader election in distributed computing. Based on this connection, the author managed to construct the first explicit extractor that uses only a constant number of sources even if the entropy is as small as $\mathrm{polylog}(n)$ [2]. More specifically, for any constant $\eta > 0$, the result gives an explicit extractor for min-entropy $k \geq \log^{2+\eta} n$ that uses $O(\frac{1}{\eta}) + O(1)$ independent (n, k) sources.

However, the result in [2] still suffers from two drawbacks. First, the $O(1)$ term can be pretty large. This is because the construction first uses a seeded extractor to convert several independent (n, k) sources into somewhere random sources (by using every possible value of the seed to extract from the source and then taking the concatenation), and then takes the XOR of these somewhere random sources to reduce the error. To ensure efficiently computability we need the seed length of the seeded extractor to be $O(\log n)$; while to ensure the number of sources needed is a constant, we need the error of the seeded extractor to be at most $1/\mathrm{poly}(n)$. Thus, we need an optimal (up to constant factors) seeded extractor in the case where the error $\epsilon = 1/\mathrm{poly}(n)$.

Suppose we have a seeded extractor with seed length $d = \log n + C \log(1/\epsilon)$ for some constant $C > 1$, then the above XOR step needs at least $C + 1$ independent weak sources. Radhakrishnan and Ta-Shma [20] showed that the constant C here must be at least 2, thus even if we have truly optimal seeded extractors, this step requires at least 3 sources. After that we need at least one extra source to convert the somewhere random source into another somewhere random source with the "almost h-wise independent property" as in [2], and we need at least two other sources to extract nearly uniform random bits. Therefore, even with truly optimal seeded extractors the construction in [2] requires at least 6 independent sources.

Unfortunately, currently we do not have truly optimal seeded extractors, but rather extractors that are optimal up to constant factors. The two known constructions of such extractors are [5] and [6] (and the related [7]). Using these seeded extractors, the $O(1)$ term in the result of [2] can be even larger (e.g., ≥ 30).

Another drawback of the result in [2] is that the construction only achieves error $1/\mathrm{poly}(n)$. This kind of error is not enough for many cryptographic applications, where we typically need to have a negligible error (i.e., $n^{-\omega(1)}$).

A. Our results

In this paper, we further improve the results in [2]. We construct an explicit extractor for three independent (n, k) sources with min-entropy $k \geq \mathrm{polylog}(n)$. In fact, our extractor works for one source with poly-logarithmic min-entropy and another independent block source with two blocks each having poly-logarithmic min-entropy. Our results are thus nearly optimal. We also improve the error of the extractor from $1/\mathrm{poly}(n)$ to $2^{-k^{\Omega(1)}}$. Specifically, we have the following theorem.

Theorem I.4. *For all $n, k \in \mathbb{N}$ with $k \geq \log^{12} n$, there is an efficiently computable function $\mathsf{IExt} : \{0,1\}^n \times \{0,1\}^{2n} \to \{0,1\}^m$ such that if X is an (n, k)-source and $Y = (Y_1, Y_2)$ is an independent (k, k) block source where each block has n bits, then*

$$|(\mathsf{IExt}(X, Y), Y) - (U_m, Y)| \leq \epsilon$$

and

$$|(\mathsf{IExt}(X, Y), X) - (U_m, X)| \leq \epsilon,$$

where $m = 0.9k$ and $\epsilon = 2^{-k^{\Omega(1)}}$. [1]

[1] We can show that this error is strictly $n^{-\omega(1)}$.

As a corollary this immediately gives the following theorem.

Theorem I.5. *For all $n, k \in \mathbb{N}$ with $k \geq \log^{12} n$, there is an efficiently computable three-source extractor $\mathsf{IExt} : (\{0,1\}^n)^3 \to \{0,1\}^m$ such that if X, Y, Z are three independent (n,k)-sources, then*

$$|\mathsf{IExt}(X,Y,Z) - U_m| \leq \epsilon,$$

where $m = 0.9k$ and $\epsilon = 2^{-k^{\Omega(1)}}$.

If the min-entropy k is very close to $\log^2 n$, then we also have improved results over [19]. In particular, we have the following theorem.

Theorem I.6. *For every constant $\eta > 0$ and all $n, k \in \mathbb{N}$ with $k \geq \log^{2+\eta} n$, there is an efficiently computable extractor $\mathsf{BExt} : (\{0,1\}^n)^t \times (\{0,1\}^n)^t \to \{0,1\}^m$ with $t = \lceil \frac{7}{\eta} \rceil + 1$, such that if $X = (X_1, X_2, \cdots X_t), Y = (Y_1, Y_2, \cdots Y_t)$ are two independent $(k, k, \cdots, k)$- block sources where each block has n bits, then*

$$|(\mathsf{BExt}(X,Y), Y) - (U_m, Y)| \leq \epsilon$$

and

$$|(\mathsf{BExt}(X,Y), X) - (U_m, X)| \leq \epsilon,$$

where $m = 0.9k$ and $\epsilon = 2^{-k^{\Omega(1)}}$.

As a corollary, we immediately obtain the following theorem.

Theorem I.7. *For every constant $\eta > 0$ and all $n, k \in \mathbb{N}$ with $k \geq \log^{2+\eta} n$, there is an efficiently computable extractor $\mathsf{IExt} : (\{0,1\}^n)^t \to \{0,1\}^m$ with $t = \lceil \frac{14}{\eta} \rceil + 2$ such that if $X_1, \cdots, X_t$ are t independent (n,k)-sources, then*

$$|\mathsf{IExt}(X_1, \cdots, X_t) - U_m| \leq \epsilon,$$

where $m = 0.9k$ and $\epsilon = 2^{-k^{\Omega(1)}}$.

For example, the above theorem gives an extractor for min-entropy $k = \log^3 n$ that uses 16 sources, and an extractor for min-entropy $k = \log^4 n$ that uses 9 sources.

Remark I.8. In all theorems, the constant 0.9 can be replaced by any constant less than 1.

Table I summarizes our results compared to previous constructions of independent source extractors.

II. Overview of The Constructions and Techniques

Here we give a brief overview of our constructions and the techniques. To give a clear description of the ideas, we shall be informal and imprecise sometimes.

The high level framework of our constructions follows that of [2], [19]. Thus, we first briefly review the construction in [2].

A. A brief review of the construction in [2]

The constant-source extractor in [2] works by first obtaining a somewhere random source (SR-source for short), which is a random $N \times m$ matrix such that at least one row is uniform. In addition, the SR-source has the stronger property that say $\frac{2}{3}$ of the rows are uniform, and moreover they are (almost) h-wise independent with $h = k^\alpha$ for some constant $0 < \alpha < 1$. Once we have this SR-source, we can use the lightest bin protocol from [21] to reduce the number of rows in the SR-source; while after each execution of the lightest bin protocol, we use the random strings in the output of the protocol as seeds to extract from another fresh weak source, using a strong seeded extractor. This way we can ensure that the resulted new random variable (not the strings from the original SR-source) is another SR-source that preserves the h-wise independent property (as long as the output length of the seeded extractor is small, say at most $k/(2h)$). On the other hand the number of rows in this new SR-source has decreased a lot, roughly from N to $N^{4/\sqrt{h}}$.

Construction	Number of Sources	Min-Entropy	Output	Error
[10]	2	$k \geq (1/2+\delta)n$, any constant δ	$\Theta(n)$	$2^{-\Omega(n)}$
[14]	$\mathrm{poly}(1/\delta)$	δn, any constant δ	$\Theta(n)$	$2^{-\Omega(n)}$
[15]	3	δn, any constant δ	$\Theta(1)$	$O(1)$
[16]	3	One source: δn, any constant δ. Other sources may have $k \geq \mathrm{polylog}(n)$.	$\Theta(1)$	$O(1)$
[16]	2	One source: $(1/2+\delta)n$, any constant δ. Other source may have $k \geq \mathrm{polylog}(n)$	$\Theta(k)$	$2^{-\Omega(k)}$
[1]	2	$(1/2-\alpha_0)n$ for some small universal constant $\alpha_0 > 0$	$\Theta(n)$	$2^{-\Omega(n)}$
[17]	3	One source: δn, any constant δ. Other sources may have $k \geq \mathrm{polylog}(n)$.	$\Theta(k)$	$2^{-k^{\Omega(1)}}$
[17]	$O(\log n/\log k)$	$k \geq \mathrm{polylog}(n)$	$\Theta(k)$	$k^{-\Omega(1)}$
[11]	$O(\log n/\log k)$	$k \geq \mathrm{polylog}(n)$	$\Theta(k)$	$2^{-k^{\Omega(1)}}$
[18]	3	$k = n^{1/2+\delta}$, any constant δ	$\Theta(k)$	$k^{-\Omega(1)}$
[19]	$O(\log(\frac{\log n}{\log k}))+O(1)$	$k \geq \mathrm{polylog}(n)$	$\Theta(k)$	$k^{-\Omega(1)}$
[2]	$O(\frac{1}{\eta})+O(1)$, $O(1)$ can be large	$k \geq \log^{2+\eta} n$	$\Theta(k)$	$n^{-\Omega(1)}+2^{-k^{\Omega(1)}}$
This work	3	$k \geq \log^{12} n$	$\Theta(k)$	$2^{-k^{\Omega(1)}}$
This work	$\lceil \frac{14}{\eta} \rceil + 2$	$k \geq \log^{2+\eta} n$	$\Theta(k)$	$2^{-k^{\Omega(1)}}$

Table I
SUMMARY OF RESULTS ON EXTRACTORS FOR INDEPENDENT SOURCES.

We can thus repeat this process until the number of rows in the SR-source becomes small enough, say $k^{1/3}$; and then we can take at most two other independent (n,k) sources and use an extractor from [11] to extract nearly uniform random bits. Since initially the number of rows in the SR-source is $\mathrm{poly}(n)$, $k \geq \mathrm{polylog}(n)$ and $h = k^{\alpha}$, a simple calculation shows that the number of iterations needed is a constant. In addition, the initial SR-source can also be obtained from a constant number of independent (n,k) sources. Thus the total number of sources needed is a constant. However, as mentioned before, the step of obtaining the initial SR-source may require a large constant number of sources.

B. The new construction

We now describe our new construction. Again, we will first obtain an SR-source such that say $\frac{2}{3}$ of the rows are uniform, and moreover they are (almost) h-wise independent with $h = k^{\alpha}$ for some constant $0 < \alpha < 1$. However, we will use just *two* independent (n,k) sources to achieve this. This is our major improvement over the construction in [2]. To explain the ideas, we will first show how to use *three* independent (n,k) sources to obtain the SR-source.

1) Use three sources to obtain the h-wise independent SR-source: In [2], the initial SR-source with the h-wise independent property is obtained in two steps. First, one uses a constant number of independent (n,k) sources to obtain a random variable that is *statistically close* to an SR-source such that say $\frac{2}{3}$ of the rows are uniform (but without the h-wise independent property). Then one can use a single extra independent (n,k) source to obtain a new SR-source with the h-wise independent property. It is the first step that uses a large number of independent sources. The reason is that if we take a seeded extractor with seed length $d = \log n + C\log(1/\epsilon)$ for some $\epsilon = 1/\mathrm{poly}(n)$ and convert a weak source into a somewhere (close to) random source by trying all possible values of the seed and then concatenating the outputs, then the number of rows is $N = 2^d > (1/\epsilon)^C$. In addition, the best one can say about the close to uniform rows is that each one is ϵ-close to uniform (or even worse). Thus if we want the source to be statistically close to an SR-source such that $\frac{2}{3}N$ rows are simultaneously uniform, by the union bound we would need the error of the close to uniform rows to be smaller than ϵ^C. Thus, it takes the XOR of at least $C+1$ independent sources applied with the seeded extractor to reduce the error to this small.

Here we take a completely different approach. Since eventually we need the error of the close to uniform rows in the source (obtained by applying a seeded extractor to an (n,k) source X and trying all possible values of the seed) to be small, we might as well just start with a seeded extractor with larger seed length, say $\ell = k^{\beta} \gg \log n$, where $0 < \beta < 1$ is another constant. Now if we use an optimal strong seeded extractor Ext_2 such as that in [5],

we can indeed show that the error of the close to uniform rows is $\epsilon = 2^{-\Omega(k^\beta)}$, which is small enough. Moreover, by a standard averaging argument we can show that at least 0.9 fraction of the rows are ϵ-close to uniform.

However, by naively doing this, we have increased the number of rows in the somewhere (close to) uniform source (which we will call $\bar{X}$) to $2^\ell = 2^{k^\beta}$, which is super polynomial and also much larger than $1/\epsilon$, so it seems that we have gained nothing. Fortunately, so far we have just used one weak source. Thus we can take another weak source and use it to *sample* a subset of $\mathrm{poly}(n)$ rows from $\bar{X}$, and hopefully with high probability conditioned on the second source, the sampled subset of rows still contains a large fraction of close to uniform rows. If this is true then we are done, since now we only have $\mathrm{poly}(n)$ rows and the error of each close to uniform row is $\epsilon = 2^{-\Omega(\ell)} = 2^{-\Omega(k^\beta)} \ll 1/\mathrm{poly}(n)$; so we can show that this new source is $\mathrm{poly}(n)2^{-\Omega(k^\beta)} = 2^{-k^{\Omega(1)}}$-close to an SR-source such that say $\frac{2}{3}$ of the rows are uniform.

Given this idea, it is straightforward to implement it. To sample from a set of elements using a weak random source, it suffices to take a seeded extractor, which is equivalent to a sampler as shown in [22]. More specifically, take a seeded $(k' = k/2, \epsilon')$ extractor Ext_1 with seed length $d = O(\log n + \log(1/\epsilon'))$ and output length $\ell = k^\beta < 0.4k$ such as that in [5], we can view it as a bipartite graph with 2^n vertices on the left, 2^ℓ vertices on the right, and left degree 2^d. Thus each vertex on the left selects a subset of right vertices with size 2^d. Now if we associate the right vertices with the 2^ℓ rows in $\bar{X}$, we can use another independent (n, k) source Y to sample a vertex on the left, which gives us a subset of the rows in $\bar{X}$ with size 2^d.

We say a row in $\bar{X}$ is "good" if it is ϵ-close to uniform. Thus at least 0.9 fraction of the rows are good. A standard property of the (k', ϵ') seeded extractor implies that the number of left vertices whose induced subset of rows in $\bar{X}$ contains less than $0.9 - \epsilon'$ fraction of good rows, is at most $2^{k'}$. Since Y is an (n, k) source, the probability of selecting a subset of rows which contains at least $0.9 - \epsilon'$ fraction of good rows is at least $1 - 2^{k'}2^{-k} = 1 - 2^{-k/2}$. Thus it suffices to take $\epsilon' = 1/4$ and we know that with probability at least $1 - 2^{-k/2}$ over Y, the selected subset of rows of $\bar{X}$ has at least $0.9 - 1/4 > 2/3$ fraction of good rows. Moreover, since $\epsilon' = 1/4$ we have that $d = O(\log n + \log(1/\epsilon')) = O(\log n)$, therefore the size of the selected subset is $2^d = \mathrm{poly}(n)$.

Note that the above sampling process is equivalent to computing $\mathsf{Ext}_2(X, \mathsf{Ext}_1(Y, r_i))$ for all possible values r_i of the d bit seed of Ext_1. Thus (although we are sampling from a set of super-polynomial size) this can be done in polynomial time. Hence, we have used two independent (n, k) sources to obtain a new source W such that with high probability, W is statistically close to an SR-source which has $\frac{2}{3}$ fraction of uniform rows. We can now take another independent source Z and use the method in [2] to get an SR-source with the h-wise independent property.

Furthermore, notice that by doing this we have reduced the error from $1/\mathrm{poly}(n)$ in [2] to $2^{-k^{\Omega(1)}}$. Essentially, with one source we can only obtain an SR-source with $\mathrm{poly}(n)$ rows such that some rows are $1/\mathrm{poly}(n)$-close to uniform; but with two independent sources we can obtain an SR-source with $\mathrm{poly}(n)$ rows such that some rows are $2^{-k^{\Omega(1)}}$ (or even $2^{-\Omega(k)}$)-close to uniform. In fact, this method is quite general and can be applied to any construction that involves reducing the error in an SR-source. For example, it can also be used to reduce the error of the extractor in [17] from $1/\mathrm{poly}(n)$ to $2^{-k^{\Omega(1)}}$. On the other hand, the method used in [11] to reduce the error of the extractor in [17] cannot be directly applied to the construction in [2], since here we need to deal with $\mathrm{poly}(n)$ random rows instead of just one random row as in [11], [17].

2) Use two sources to obtain the h-wise independent SR-source: We now describe how we can remove one source, and use just two independent (n, k) sources to obtain the h-wise independent SR-source. First, we also briefly review the method to generate the h-wise independent SR-source in [2]. Given an SR-source Y and an independent source X, we will use each row of Y to do several rounds of alternating extraction (cf. [23]–[26]) from X. More specifically, we divide the binary expression of the index of the row of Y into blocks of size $\log h$, and for each block we run an alternating extraction from X and pick an output indexed by that block. This output is then used to start the next round of alternating extraction. The final output will be the output of the alternating extraction in the last round, indexed by the last block of the binary expression of the index of that row (more details can be found in [2]). The new SR-source Z will then be the concatenation of the outputs for all rows.

In each alternating extraction the seed length of the seeded extractor is chosen to be $\ell = k^\beta$, and one can show the following. For any subset of rows in Y with size h, if all these rows are uniform (but they may depend on each other arbitrarily), then with probability $1 - 2^{-\Omega(\ell)}$ over the fixing of Y, the joint distribution of the corresponding rows in Z is $2^{-\Omega(\ell)}$-close to uniform (i.e., Z has the almost h-wise independent property).

Now, going back to our new construction. We have already used two independent sources Y and X to obtain an

SR-source W with $N = \text{poly}(n)$ rows, such that with probability $1 - 2^{-k/2}$ over the fixing of Y, there exists a large subset $T \subseteq [N]$ such that each row of W with index in T is $2^{-\Omega(\ell)}$-close to uniform. Moreover we will have Ext_2 output ℓ bits so that each row in W has length ℓ. We will now take another optimal seeded extractor, and then use each row of W as the seed to extract from Y and output $k/2$ bits. Let the concatenation of these outputs be $\bar{Y}$. We will now think of $\bar{Y}$ as an SR-source, and X as an independent source, and use the same method in [2] described above to obtain the new SR-source Z from $\bar{Y}$ and X.

We will show that with high probability over the fixing of Y, the new SR-source Z has the desired h-wise independent property. Note that with probability $1-2^{-k/2}$ over the fixing of Y, there exists a large subset $T \subseteq [N]$ such that each row of W with index in T is $2^{-\Omega(\ell)}$-close to uniform. If for every $y \in \mathsf{Supp}(Y)$ that makes this happen, we can show that conditioned on $Y = y$, the new source Z also has the desired h-wise independent property in the subset T of rows then we are done. However, this may not be the case. Thus, we want to subtract from $1-2^{-k/2}$ the probability mass of the "bad" y's which result in a Z that does not have the h-wise independent property in the subset T of rows. Towards this goal, we define a bad $y \in \mathsf{Supp}(Y)$ to be a string that satisfies the following two properties:

a) **Conditioned on the fixing of $Y = y$, there exists a large subset $T \subseteq [N]$ such that each row of W with index in T is $2^{-\Omega(\ell)}$-close to uniform**,
and
b) **Conditioned on the fixing of $Y = y$, there exists a subset $S \subseteq T$ with $|S| = h$ such that the joint distribution of the rows of Z with index in S is ϵ_1 far from uniform**, where ϵ_1 is an error parameter to be chosen later.

Note that $S \subseteq T$, since y satisfies condition a), we must have that conditioned on the fixing of $Y = y$, each row of W with index in S is $2^{-\Omega(\ell)}$-close to uniform. Therefore, for each $S \subseteq [N]$ with $|S| = h$ we now define an event Bad_S to be the set of y's in $\mathsf{Supp}(Y)$ that satisfies the following two properties:

c) **Conditioned on the fixing of $Y = y$, each row of W with index in S is $2^{-\Omega(\ell)}$-close to uniform**,
and
d) **Conditioned on the fixing of $Y = y$, the joint distribution of the rows of Z with index in S is ϵ_1 far from uniform**.

Thus every bad y must belong to some Bad_S. Therefore to bound the probability mass of the bad y's we only need to bound $\Pr[Bad_S]$ for every S and then take a union bound. Now the crucial observation is that for any fixed subset S, property c) is determined by the h random variables $R_i = \mathsf{Ext}_1(Y, r_i)$ with $i \in S$. Let R be the concatenation of $\{R_i, i \in S\}$ (which is a deterministic function of Y), and define the event A_S to be the set of r's in $\mathsf{Supp}(R)$ that makes property c) satisfied, then we have $\Pr[Bad_S] = \sum_{r \in A_S} \Pr[R = r] \Pr[Bad_S | R = r]$.

Now another crucial observation is that the size of R is small. Indeed, it is bounded by $h\ell = k^{\alpha+\beta}$. If we choose α, β to be such that $\alpha + \beta < 1$, then the size of R is $o(k)$ and we can argue that with probability $1 - 2^{-\ell}$ over the fixing of $R = r$, we have that Y still has min-entropy at least $k - o(k) - \ell = k - o(k) > 0.9k$. Moreover conditioned on the fixing of $R = r$ we have that $\{W_i, i \in S\}$ is a deterministic function of X, and is thus independent of Y.

We now bound $\Pr[Bad_S | R = r]$ in two cases. First, if $H_\infty(Y|R = r) < 0.9k$, we will just use $\Pr[Bad_S | R = r] \leq 1$. By the above argument this happens with probability at most $2^{-\ell}$. We now consider the case where $H_\infty(Y|R = r) \geq 0.9k$. In this case, we know that for all $i \in S$, W_i is $2^{-\Omega(\ell)}$-close to uniform. Thus the joint distribution of $\{W_i, i \in S\}$ is $h2^{-\Omega(\ell)} = 2^{-\Omega(\ell)}$-close (since $h = k^\alpha$ and $\ell = k^\beta$) to a source with h truly uniform rows. Ignoring the error for the moment, we can now say that for all $i \in S$, $|(\bar{Y}_i, W_i) - (U_\ell, W_i)| \leq 2^{-\Omega(\ell)}$. Thus for all $i \in S$, with probability $1 - 2^{-\Omega(\ell)}$ over the fixing of W_i, we have that $\bar{Y}_i$ is $2^{-\Omega(\ell)}$-close to uniform. This implies that with probability $1 - h2^{-\Omega(\ell)} = 1 - 2^{-\Omega(\ell)}$ over the fixing of $\{W_i, i \in S\}$, we have that the joint distribution of $\{\bar{Y}_i, i \in S\}$ is $h2^{-\Omega(\ell)} = 2^{-\Omega(\ell)}$-close to a source with h truly uniform rows. Moreover, notice that the size of $\{W_i, i \in S\}$ is also bounded by $h\ell = k^{\alpha+\beta}$. Thus again we can argue that with probability $1 - 2^{-\ell}$ over the fixing of $\{W_i, i \in S\}$, we have that X still has min-entropy at least $k - o(k) - \ell > 0.9k$. Altogether, this implies that with probability $1 - 2^{-\Omega(\ell)} - 2^{-\ell} = 1 - 2^{-\Omega(\ell)}$ over the fixing of $\{W_i, i \in S\}$, we have that the joint distribution of $\{\bar{Y}_i, i \in S\}$ is $2^{-\Omega(\ell)}$-close to a source with h truly uniform rows, and X still has min-entropy at

least $0.9k$. In addition, after this further fixing of $\{W_i, i \in S\}$, we have that $\{\bar{Y}_i, i \in S\}$ is a deterministic function of Y, and is thus independent of X.

We can now use the same argument in [2] (treat $\{\bar{Y}_i, i \in S\}$ as the SR-source and X as an independent weak source) to argue that with probability $1 - 2^{-\Omega(\ell)}$ over the fixing of $\{\bar{Y}_i, i \in S\}$ (and thus also the fixing of Y, since $\{\bar{Y}_i, i \in S\}$ is now a deterministic function of Y), we have that the joint distribution of $\{Z_i, i \in S\}$ is $2^{-\Omega(\ell)}$-close to uniform. Now adding back all the errors, the above statement is still true (except for a slight change of constants in $\Omega(\cdot)$). Thus, if we set ϵ_1 to be some $2^{-\Omega(\ell)}$ appropriately, then we have that in this case $\Pr[Bad_S|R = r] \leq 2^{-\Omega(\ell)}$. Therefore, by combining the two cases, we get that $\Pr[Bad_S] \leq 2^{-\ell} + \Pr[A_S]2^{-\Omega(\ell)} \leq 2^{-\Omega(\ell)}$.

Now by the union bound we know the probability mass of the bad y's is at most $\binom{N}{h}2^{-\Omega(\ell)} \leq N^h 2^{-\Omega(\ell)} = 2^{O(h \log n) - \Omega(\ell)}$. If we choose α, β such that $k^{\beta-\alpha} \geq C \log n$ for some large enough constant $C > 1$, then we get that this probability mass is again $2^{-\Omega(\ell)}$. Also, by choosing the constant C appropriately, this will also ensure that the error of the h-wise independent rows (which is $2^{-\Omega(\ell)}$) is less than N^{-6h}. This will be enough for the lightest bin protocol to work, as shown in [2]. All these requirements, as well as other requirements in obtaining the h-wise independent SR-source, can be satisfied as long as $k = \log^{2+\eta} n$ for any constant $\eta > 0$ (see Algorithm **??**).

Now we are done. Subtracting the probability mass of the bad y's from $1 - 2^{-k/2}$, we get that with probability $1 - 2^{-\Omega(\ell)}$ over the fixing of Y, the source Z has the desired h-wise independent property.

3) Achieving a three-source extractor: Now that we have used two independent sources to obtain an SR-source with the h-wise independent property, we can use the rest of the construction in [2] to get an extractor. However, the direct use of the construction in [2] requires at least two more sources. This is because the lightest bin protocol requires at least one round, and at the end of that round we need to use a fresh source to get another SR-source. We then need to take another source in order to finish extraction. This will give us a four-source extractor.

In order to save one source, we observe that if the entropy k is a large enough polynomial in $\log n$, then $h = k^\alpha$ will also be large enough so that in just one iteration of the lightest bin protocol, the number of rows in the SR-source will decrease from $N = \text{poly}(n)$ to say $N' \leq k^{1/3}$. We let the concatenation of these rows of Z be Z'. Note that Z' is a deterministic function of Z. By cutting the length of each row of Z (if necessary) to say $\sqrt{k}$, we see that the size of Z' is bounded by $N'\sqrt{k} \leq k^{5/6}$. At the end of the lightest bin protocol we will take a fresh weak source Y_2 (this is the third source) and use each row of Z' to extract a string of length say $0.9k$ from Y_2 (by using an optimal seeded extractor). We let the concatenation of these outputs be Y'. The analysis in [2] implies that with high probability over the fixing of Z, the new source Y' is also (close to) an SR-source (here it is not necessary to have the h-wise independent property).

Note that Y' is a deterministic function of Y_2 and Z', and Z' is deterministic function of Z. Thus it is also true that with high probability over the fixing of Z', the new source Y' is close to an SR-source. Moreover conditioned on the fixing of Y, we have that Z is a deterministic function of X. Since the size of Z' is $o(k)$, we can argue that with high probability over the fixing of Z', the min-entropy of X is $k - o(k) > 0.9k$. Furthermore, conditioned on the fixing of (Y, Z'), we have that X and Y' are independent. Note that Y' is an SR-source with $k^{1/3}$ rows but each row has length $0.9k \gg k^{1/3}$, thus by using an extractor from [11] we can extract random bits from X and Y' which are $2^{-k^{\Omega(1)}}$-close to uniform. This gives our three-source extractor with error $2^{-k^{\Omega(1)}}$. It turns out that it is enough to choose $k \geq \log^{12} n$ and $\alpha = 1/6, \beta = 1/3$ in this case. Also notice here that Y and Y_2 need not be independent, but rather it suffices to have (Y, Y_2) be a block source (since the analysis first conditions on the fixing of Y). Thus our construction actually gives an extractor for one (n, k) source and another independent (k, k)-block source (see Algorithm V.9).

4) Improving the results of [2] for smaller min-entropy: Our three-source extractor requires $k \geq \log^{12} n$. However, if $k = \log^{2+\eta} n$ for some small constant $\eta > 0$, then we can also get improved results by replacing the step of obtaining the h-wise independent SR-source in [2] with our new construction, which uses only two independent sources. This way we get a constant-source extractor with error $2^{-k^{\Omega(1)}}$.

Moreover, once we have this SR-source, running the lightest bin protocol actually does not need fully independent sources. For example, if $X = (X_1, \cdots, X_t)$ and $Y = (Y_1, \cdots, Y_t)$ are two independent block sources where each block has min-entropy k conditioned on all previous blocks, then we can first obtain the SR-source Z from (X_1, Y_1). Now we know that with high probability conditioned on the fixing of Y_1, the source Z has the desired property; moreover it is a deterministic function of X. Thus we can run the lightest bin protocol once and take a new block from Y to obtain a new SR-source Z_2, which is a deterministic function of Y conditioned on Z; we can then run the lightest bin protocol again and take a new block from X to obtain a new SR-source Z_3, which is a deterministic

function of X conditioned on Z_2, and so on. This gives us an extractor for two independent block sources with each having a constant number of blocks.

Subsequent work. Following the appearance of our work online, there have been several excellent follow-up works. Based on the techniques in one of the author's previous work [2], Cohen [27] found a way to improve the part in [2] about generating the h-wise independent SR-source from one SR-source and another independent weak source. As a result, our three source extractor can be improved to work for smaller min-entropy (a rough estimation gives $k \geq \log^7 n$). He also constructed an extractor for three independent sources with entropy δn, $O(\log n)$ and $O(\log \log n)$ respectively. In a separate work, using our extractor for one independent source and another block source as a key ingredient, Cohen [28] constructed the first two-source disperser for polylograithmic min-entropy, and thus giving the first explicit $2^{(\log \log n)^c}$-Ramsey graphs.

Notice that our construction actually converts two independent sources into a non-oblivious bit-fixing source where the "good" bits are almost t-wise independent for some $t = k^{\Omega(1)}$. In a very recent breakthrough [29], Chattopadhyay and Zuckerman gave an explicit deterministic extractor for such sources (they used slightly different techniques to obtain the non-oblivious bit-fixing source), and thus obtaining the first explicit two-source extractor for min-entorpy $k \geq \mathrm{polylog}(n)$. However, their construction only outputs one bit. This was later improved by the author in [30] to give an explicit two-source extractor for min-entorpy $k \geq \mathrm{polylog}(n)$ that can output $k^{\Omega(1)}$ bits if the extractor is strong, and one that can output k bits if the extractor is not necessarily strong.

Organization. The rest of the paper is organized as follows. We give some preliminaries in Section III. In Section IV we define alternating extraction, an important ingredient in our construction. We present our main construction of extractors in Section V.

III. Preliminaries

We often use capital letters for random variables and corresponding small letters for their instantiations. Let $|S|$ denote the cardinality of the set S. For ℓ a positive integer, U_ℓ denotes the uniform distribution on $\{0,1\}^\ell$. When used as a component in a vector, each U_ℓ is assumed independent of the other components. All logarithms are to the base 2.

A. Probability distributions

Definition III.1 (statistical distance)**.** Let W and Z be two distributions on a set S. Their *statistical distance* (variation distance) is

$$\Delta(W,Z) \stackrel{def}{=} \max_{T \subseteq S}(|W(T) - Z(T)|) = \frac{1}{2}\sum_{s \in S}|W(s) - Z(s)|.$$

We say W is ε-close to Z, denoted $W \approx_\varepsilon Z$, if $\Delta(W,Z) \leq \varepsilon$. For a distribution D on a set S and a function $h : S \to T$, let $h(D)$ denote the distribution on T induced by choosing x according to D and outputting $h(x)$.

B. Somewhere Random Sources and Extractors

Definition III.2 (Somewhere Random sources)**.** 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.

Definition III.3. (Block Sources) A distribution $X = X_1 \circ X_2 \circ \cdots, \circ X_t$ is called a $(k_1, k_2, \cdots, k_t)$ block source if for all $i = 1, \cdots, t$, we have that for all $x_1 \in \mathsf{Supp}(X_1), \cdots, x_{i-1} \in \mathsf{Supp}(X_{i-1})$, $H_\infty(X_i | X_1 = x_1, \cdots, X_{i-1} = x_{i-1}) \geq k_i$, i.e., each block has high min-entropy even conditioned on any fixing of the previous blocks. If $k_1 = k_2 = \cdots = k_t = k$, we say that X is a k block source.

C. Prerequisites from previous work

For a strong seeded extractor with optimal parameters, we use the following extractor constructed in [5].

Theorem III.4 ([5])**.** *For every constant $\alpha > 0$, and all positive integers n, k and any $\epsilon > 0$, there is an explicit construction of a strong (k, ϵ)-extractor $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ with $d = O(\log n + \log(1/\epsilon))$ and $m \geq (1-\alpha)k$.*

Theorem III.5 ([11]). *For every $n, k(n)$ with $k > \log^2 n$, and any constants $0 < \eta < 1$, $0 < \gamma < 1/2$ such that $k^{1-2\gamma} \geq \log^{1.1} n$, there exist constants $0 < \alpha, \beta < 1$ and 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, $m = (1-\eta)k$ and $\epsilon = 2^{-k^\alpha}$.

Remark III.6. The original version of [11] requires $k > \log^{10} n$. But this is only because the output length is $m = k - k^{\Omega(1)}$, and to achieve such output length, currently the best known seeded extractor requires seed length $d = O(\log^3(n/\epsilon))$. If we only need to achieve output length $m = (1-\eta)k$, then we can use a seeded extractor with seed length $d = O(\log(n/\epsilon))$, such as [5]. Then it suffices to have $k > \log^2 n$ for some properly chosen α, β.

The following standard lemma about conditional min-entropy is implicit in [3] and explicit in [31].

Lemma III.7 ([31]). *Let X and Y be random variables and let $\mathcal{Y}$ denote the range of Y. Then for all $\epsilon > 0$, one has*

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

We also need the following lemma.

Lemma III.8 ([26]). *Let (X,Y) be a joint distribution such that X has range $\mathcal{X}$ and Y has range $\mathcal{Y}$. Assume that there is another random variable X' with the same range as X such that $|X - X'| = \epsilon$. Then there exists a joint distribution (X', Y) such that $|(X,Y) - (X',Y)| = \epsilon$.*

IV. Alternating Extraction

As in [2], an important ingredient in the construction of our extractors is the following alternating extraction protocol.

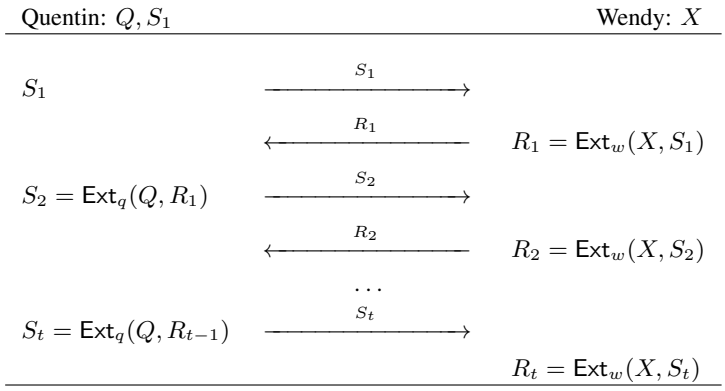

Figure 1. Alternating Extraction.

Alternating Extraction. Assume that we have two parties, Quentin and Wendy. Quentin has a source Q, Wendy has a source X. Also assume that Quentin has a uniform random seed S_1 (which may be correlated with Q). Suppose that (Q, S_1) is kept secret from Wendy and X is kept secret from Quentin. Let Ext_q, Ext_w be strong seeded extractors with optimal parameters, such as that in Theorem III.4. Let ℓ be an integer parameter for the protocol. For some integer parameter $t > 0$, the *alternating extraction protocol* is an interactive process between Quentin and Wendy that runs in t steps.

In the first step, Quentin sends S_1 to Wendy, Wendy computes $R_1 = \mathsf{Ext}_w(X, S_1)$. She sends R_1 to Quentin and Quentin computes $S_2 = \mathsf{Ext}_q(Q, R_1)$. In this step R_1, S_2 each outputs ℓ bits. In each subsequent step i,

Quentin sends S_i to Wendy, Wendy computes $R_i = \mathsf{Ext}_w(X, S_i)$. She replies R_i to Quentin and Quentin computes $S_{i+1} = \mathsf{Ext}_q(Q, R_i)$. In step i, R_i, S_{i+1} each outputs ℓ bits. Therefore, this process produces the following sequence:

$$S_1, R_1 = \mathsf{Ext}_w(X, S_1), S_2 = \mathsf{Ext}_q(Q, R_1), \cdots, S_t = \mathsf{Ext}_q(Q, R_{t-1}), R_t = \mathsf{Ext}_w(X, S_t).$$

Look-Ahead Extractor. Now we can define our look-ahead extractor. Let $Y = (Q, S_1)$ be a seed, the look-ahead extractor is defined as

$$\mathsf{laExt}(X, Y) = \mathsf{laExt}(X, (Q, S_1)) \stackrel{def}{=} R_1, \cdots, R_t.$$

The following lemma is proved in [2].

Lemma IV.1. *Let $Y = (Q, S_1)$ where Q is an (n_q, k_q) source and S_1 is the uniform distribution over ℓ bits. Let $Y_2 = (Q_2, S_{21}), \cdots, Y_h = (Q_h, S_{h1})$ be another $h-1$ random variables with the same range of Y that are arbitrarily correlated to Y. Assume that X is an (n, k) source independent of $(Y, Y_2, \cdots, Y_h)$, such that $k > ht\ell + 10\ell + 2\log(1/\epsilon)$ and $k_q > ht\ell + 10\ell + 2\log(1/\epsilon)$. Assume that Ext_q and Ext_w are strong seeded extractors that use ℓ bits to extract from $(n_q, 10\ell)$ sources and $(n, 10\ell)$ sources respectively, with error ϵ and $\ell = O(\log(max\{n_q, n\}) + \log(1/\epsilon))$. Let $(R_1, \cdots, R_t) = \mathsf{laExt}(X, Y)$ and $(R_{i1}, \cdots, R_{it}) = \mathsf{laExt}(X, Y_i)$ for $i = 2, \cdots, h$. Then for any $0 \le j \le t-1$, we have*

$$(Y, Y_2, \cdots, Y_h, \{R_{i1}, \cdots, R_{ij}, i = 2, \cdots, h\}, R_{j+1}) \approx_{\epsilon_1} (Y, Y_2, \cdots, Y_h, \{R_{i1}, \cdots, R_{ij}, i = 2, \cdots, h\}, U_\ell),$$

where $\epsilon_1 = O(t\epsilon)$.

V. The Extractor

In this section we give our main construction. We will take two parameters $0 < \alpha < \beta < 1$ and let $h \approx k^\alpha$ and $\ell = k^\beta$. The first step is to obtain an SR-source such that a large fraction of the rows are roughly h-wise independent. We have the following claim and lemma.

Claim V.1. *Let $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ be a (k, ϵ) seeded extractor. For any $T \subseteq \{0,1\}^m$ and $\rho = |T|/2^m$, let $Bad_T = \{x \in \{0,1\}^n : \Pr_{r \leftarrow U_d}[\mathsf{Ext}(x, r) \in T] > \rho + \epsilon\}$. Then*

$$|Bad_T| \le 2^k.$$

Lemma V.2. *Let $\mathsf{Ext}_1 : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ be a (k_1, ϵ_1) seeded extractor, and $\mathsf{Ext}_2 : \{0,1\}^n \times \{0,1\}^m \to \{0,1\}^{m_2}$ be a (k_2, ϵ_2) strong seeded extractor. Let Y be an $(n, 2k_1)$ source and X be an independent (n, k_2) source. For $i = 0, 1, \cdots, 2^d - 1$, let $Z_i = \mathsf{Ext}_2(X, \mathsf{Ext}_1(Y, r_i))$, where r_i is the d bit string of i's binary expression. Then with probability $1 - 2^{-k_1}$ over the fixing of Y, there exists a subset $S \subseteq \{0, 1, \cdots, 2^d - 1\}$ such that the following holds:*

- $|S| \ge (1 - \sqrt{\epsilon_2} - \epsilon_1)2^d$.
- $\forall i \in S$, *we have* $|Z_i - U_{m_2}| \le \sqrt{\epsilon_2}$.

Proof: Let R be a uniform random string over $\{0,1\}^m$. Since Ext_2 is a (k_2, ϵ_2) strong seeded extractor, we have

$$\Pr_{r \leftarrow R}[|\mathsf{Ext}_2(X, r) - U_{m_2}| \ge \sqrt{\epsilon_2}] \le \sqrt{\epsilon_2}.$$

Let $Bad_X = \{r \in \{0,1\}^m : |\mathsf{Ext}_2(X, r) - U_{m_2}| \ge \sqrt{\epsilon_2}\}$, then $|Bad_X| \le \sqrt{\epsilon_2}2^m$. Now let R' be the uniform distribution over $\{0,1\}^d$ and let $Bad_Y = \{y \in \{0,1\}^n : \Pr[\mathsf{Ext}_1(y, R') \in Bad_X] > \sqrt{\epsilon_2} + \epsilon_1\}$. Then by Claim V.1 we have that

$$|Bad_Y| \le 2^{k_1}.$$

Thus if Y is an $(n, 2k_1)$ source, then $\Pr_{y \leftarrow Y}[y \in Bad_Y] \leq 2^{k_1} 2^{-2k_1} = 2^{-k_1}$. When $y \notin Bad_Y$, we have that $\Pr[\mathsf{Ext}_1(y, R') \in Bad_X] \leq \sqrt{\epsilon_2} + \epsilon_1$, which implies that there exists a subset $S \subseteq \{0, 1, \cdots, 2^d - 1\}$ with $|S| \geq (1 - \sqrt{\epsilon_2} - \epsilon_1)2^d$ and $\forall i \in S$, $|Z_i - U_{m_2}| = |\mathsf{Ext}_2(X, \mathsf{Ext}_1(y, r_i)) - U_{m_2}| \leq \sqrt{\epsilon_2}$. ■

Suppose we have an (n, k) source X with $k \geq \text{polylog}(n)$ and an independent SR-source $Y = Y^1 \circ \cdots \circ Y^N$ with $N = \text{poly}(n)$ rows and each row has $0.9k$ bits, such that a large fraction of the rows are uniform. The following algorithm from [2] takes X and Y as inputs and outputs another SR-source Z such that a large fraction of the rows are roughly h-wise independent.

Algorithm V.3 ($\mathsf{SSR}(X, Y)$ [2])**.**

Input: X— an (n, k)-source with $k \geq \text{polylog}(n)$. $Y = Y^1 \circ \cdots \circ Y^N$—an SR-source with $N = \text{poly}(n)$ rows and each row has $0.9k$ bits, independent of X.
Output: Z — a source that is close to an SR-source.

Sub-Routines and Parameters:
Let $0 < \alpha < \beta < 1$ be the two constants above. Let $\ell = k^\beta$. Pick an integer h such that $k^\alpha \leq h < 2k^\alpha$ and $h = 2^l$ for some integer $l > 0$. Let $\mathsf{Ext}_q, \mathsf{Ext}_w$ be strong extractors with optimal parameters from Theorem III.4, set up to extract from $((h^2 + 12)\ell, 10\ell)$ sources and $(n, 10\ell)$ sources respectively, with seed length ℓ, error $\epsilon_2 = 2^{-\Omega(\ell)}$ and output length ℓ. These will be used in laExt. Let Ext be a strong extractor with optimal parameters from Theorem III.4, set up to extract from $(0.9k, 2(h^2 + 12)\ell)$ sources, with seed length ℓ, error $\epsilon_2 = 2^{-\Omega(\ell)}$ and output length $(h^2 + 12)\ell$.

1) For every $i = 1, \cdots, N$, use X and Y^i to compute Z^i as follows.
   a) Compute the binary expression of $i - 1$, which consists of $d = \log N = O(\log n)$ bits. Divide these bits sequentially from left to right into $b = \lceil \frac{d}{l} \rceil$ blocks of size l (the last block may have less than l bits, then we add 0s at the end to make it l bits). Now from left to right, for each block $j = 1, \cdots, b$, we obtain an integer $\mathsf{Ind}_{ij} \leq 2^l$ such that the binary expression of $\mathsf{Ind}_{ij} - 1$ is the same as the bits in block j.
   b) Let Y^{i1} be the first $(h + 12)\ell$ bits of Y^i. Set $j = 1$. While $j < b$ do the following.
      i) Compute $(R_1^{ij}, \cdots, R_h^{ij}) = \mathsf{laExt}(X, Y^{ij})$, where $Q = Y^{ij}$ and S_1 is the first ℓ bits of Y^{ij}.
      ii) Compute $Y^{i(j+1)} = \mathsf{Ext}(Y^i, R^{ij}_{\mathsf{Ind}_{ij}})$.
      iii) Set $j = j + 1$.
   c) Finally, compute $(R_1^{ib}, \cdots, R_h^{ib}) = \mathsf{laExt}(X, Y^{ib})$ and set $Z^i = R^{ib}_{\mathsf{Ind}_{ib}}$.
2) Let $Z = Z^1 \circ \cdots \circ Z^N$.

We now introduce some notation as in [2]. For any $i \in [N]$ and $j \in [b]$, we let $Y^{i(\leq j)}$ denote $(Y^{i1}, \cdots, Y^{ij})$, let $R^{i(\leq j)}_{\mathsf{Ind}_{i(\leq j)}}$ denote $(R^{i1}_{\mathsf{Ind}_{i1}}, \cdots, R^{ij}_{\mathsf{Ind}_{ij}})$ and let $f^j(i)$ denote the integer whose binary expression is the concatenation of the binary expression of $i - 1$ from block 1 to block j. The following lemma is proved in [2].

Lemma V.4. *Assume that $k \geq 2(bh + 2)(h^2 + 12)\ell$. Fix any $v \in [N]$ such that Y^v is uniform. Let $S \subset [N]$ be any subset with $|S| = h$ and $v \in S$. For any $j \in [b]$, define $S_v^j = \{i \in S : f^j(i) < f^j(v)\}$. Then for any $j \in [b]$, we have that*

$$(R^{vj}_{\mathsf{Ind}_{vj}}, \{Y^{i(\leq j)}, i \in S\}, \{R^{ij}_{\mathsf{Ind}_{ij}}, i \in S_v^j\}, \{R^{i(\leq j-1)}_{\mathsf{Ind}_{i(\leq j-1)}}, i \in S\})$$
$$\approx_{O(jh\epsilon_2)} (U_\ell, \{Y^{i(\leq j)}, i \in S\}, \{R^{ij}_{\mathsf{Ind}_{ij}}, i \in S_v^j\}, \{R^{i(\leq j-1)}_{\mathsf{Ind}_{i(\leq j-1)}}, i \in S\}).$$

Moreover, conditioned on the fixing of $(\{Y^{i(\leq j)}, i \in S\}, \{R^{i(\leq j-1)}_{\mathsf{Ind}_{i(\leq j-1)}}, i \in S\})$, we have that

1) *X and Y are still independent.*
2) *$(R^{ij}_{\mathsf{Ind}_{ij}}, i \in S)$ are all deterministic functions of X.*

Now we can prove the following lemma, which is slightly stronger than a similar lemma in [2].

Lemma V.5. *Assume that $k \geq 2(bh+2)(h^2+12)\ell$, X is an (n,k)-source and Y is an $N \times 0.9k$ SR-source independent of X, with $N = \mathrm{poly}(n)$ such that there exists a subset $S \subset [N]$ and for any $i \in S$, Y^i is uniform. Let $Z = Z^1 \circ \cdots \circ Z^N = \mathsf{SSR}(X,Y)$. Then for any subset $S' \subset S$ with $|S'| = h$, we have that*

$$((Z^i, i \in S'), Y) \approx_\epsilon (U_{h\ell}, Y),$$

where $\epsilon = O(bh^2\epsilon_2) = 2^{-\Omega(\ell)}$.

Proof: We order the elements in S' to be $i_1 < i_2 < \cdots < i_h$. Since $S' \subset S$, for any $j \in [h]$ we have that Y^{i_j} is uniform. We now apply Lemma V.4 to the set S' with $j = b$. Note that $f^b(i) = i - 1$, thus for any $v \in S'$ we have $S'^b_v = \{i \in S' : i < v\}$. Also note that $Z^i = R^{ib}_{\mathsf{Ind}_{ib}}$ for any $i \in [N]$. Thus by Lemma V.4, for any $j \in [h]$ we have that

$$\begin{aligned}&(Z^{i_j}, Z^{i_1}, \cdots, Z^{i_{j-1}}, \{Y^{i(\leq b)}, i \in S'\}, \{R^{i(\leq b-1)}_{\mathsf{Ind}_{i(\leq b-1)}}, i \in S'\})\\ \approx_{O(jh\epsilon_2)} &(U_\ell, Z^{i_1}, \cdots, Z^{i_{j-1}}, \{Y^{i(\leq b)}, i \in S'\}, \{R^{i(\leq b-1)}_{\mathsf{Ind}_{i(\leq b-1)}}, i \in S'\}),\end{aligned}$$

where $\epsilon_2 = 2^{-\Omega(\ell)}$.

Note that by Lemma V.4, conditioned on the fixing of $\{Y^{i(\leq b)}, i \in S'\}, \{R^{i(\leq b-1)}_{\mathsf{Ind}_{i(\leq b-1)}}, i \in S'\}$, we have that X and Y are still independent, and $(R^{ib}_{\mathsf{Ind}_{ib}}, i \in S') = (Z^i, i \in S')$ are all deterministic functions of X. Thus we also have

$$\begin{aligned}&(Z^{i_j}, Z^{i_1}, \cdots, Z^{i_{j-1}}, \{Y^{i(\leq b)}, i \in S'\}, \{R^{i(\leq b-1)}_{\mathsf{Ind}_{i(\leq b-1)}}, i \in S'\}, Y)\\ \approx_{O(jh\epsilon_2)} &(U_\ell, Z^{i_1}, \cdots, Z^{i_{j-1}}, \{Y^{i(\leq b)}, i \in S'\}, \{R^{i(\leq b-1)}_{\mathsf{Ind}_{i(\leq b-1)}}, i \in S'\}, Y),\end{aligned}$$

and therefore (since $j \leq b$)

$$(Z^{i_j}, Z^{i_1}, \cdots, Z^{i_{j-1}}, Y) \approx_{O(bh\epsilon_2)} (U_\ell, Z^{i_1}, \cdots, Z^{i_{j-1}}, Y).$$

Note this holds for every j, thus by a standard hybrid argument we have that

$$(Z^{i_1}, \cdots, Z^{i_h}, Y) \approx_\epsilon (U_{h\ell}, Y),$$

where $\epsilon = O(bh^2\epsilon_2) = O(bh^2 2^{-\Omega(\ell)}) = 2^{-\Omega(\ell)}$ since $\ell = k^\beta$, $h < 2k^\alpha$ and $b < \log n = k^{O(1)}$. ∎

We can now describe the algorithm to create an SR-source such that a large fraction of the rows are roughly h-wise independent, from just two independent sources X and Y.

Algorithm V.6 ($\mathsf{SR}(X,Y)$)**.**

Input: X, Y— two independent $(n, 2k)$-source with $k \geq \mathrm{polylog}(n)$.
Output: Z — a source that is close to an SR-source.

Sub-Routines and Parameters:
Let $0 < \alpha < \beta < 1$ be the two constants defined before. Let $\ell = k^\beta$. Let $\mathsf{Ext}_1, \mathsf{Ext}_2$ be two strong seeded extractors with optimal parameters from Theorem III.4, set up to extract from (n,k) sources. Ext_1 has seed length $d = O(\log n)$, error $\epsilon_1 = 1/4$ and output length ℓ; Ext_2 has seed length ℓ, error $\epsilon_2 = 2^{-\Omega(\ell)}$ and output length ℓ. Let Ext_3 be another strong seeded extractor with optimal parameters from Theorem III.4, set up to extract from (n,k) sources, with seed length ℓ, error ϵ_2 and output length $0.9k$ (we will choose the parameters such that $2k - (h+1)\ell \geq k$).

1) Let $N = 2^d = \mathrm{poly}(n)$. For every $i = 1, \cdots, N$, let r_i be the d bit string which is the binary expression of $i-1$. Compute $W_i = \mathsf{Ext}_2(X, \mathsf{Ext}_1(Y, r_i))$ and $Y^i = \mathsf{Ext}_3(Y, W_i)$. Let $\overline{Y} = Y^1 \circ \cdots \circ Y^N$.
2) Compute $Z = \mathsf{SSR}(X, \overline{Y})$ using Algorithm V.3.

We now have the following lemma.

Lemma V.7. *Assume that $k \geq 2(bh+2)(h^2+12)\ell$. There exists a constant $C > 1$ such that if $\ell \geq Ch \log n$, then with probability $1 - 2^{-\Omega(\ell)}$ over the fixing of Y, the following property is satisfied: there exists a subset $T \subseteq [N]$ such that $|T| \geq \frac{2}{3}N$ and $\forall S \subseteq T$ with $|S| = h$, we have*

$$|(Z_i, i \in S) - U_{h\ell}| \leq 2^{-\Omega(\ell)}.$$

Proof: Let $W = W_1 \circ \cdots \circ W_N$. We first show that with high probability over the fixing of Y, we have that W is an SR-source with a large fraction of close to uniform rows. This follows directly from Lemma V.2. Specifically, the lemma implies that with probability $1 - 2^{-k}$ over the fixing of Y, there exists a subset $T \subseteq [N]$ with $N = 2^d = \text{poly}(n)$ such that $|T| \geq (1 - \sqrt{\epsilon_2} - 1/4)N > \frac{2}{3}N$ since $\epsilon_2 = 2^{-\Omega(\ell)}$; and $\forall i \in T$, we have $|W_i - U_\ell| \leq \sqrt{\epsilon_2} = 2^{-\Omega(\ell)}$.

Now consider any $y \in \mathsf{Supp}(Y)$ which makes the above happen. We'd like to show that conditioned on this $Y = y$, in the final output Z, the same set T of the rows will also have the property of being roughly h-wise independent. However, this may not be the case; and if not, we will call such a y bad. Now fix any bad y. Then we know that there must be a subset $S \subset T$ with $|S| = h$ such that $|(Z^i, i \in S) - U_{h\ell}| > \epsilon'$ for some $\epsilon' = 2^{-\Omega(\ell)}$. At the same time, since $S \subset T$ we also know that $\forall i \in S$, we have $|W_i - U_\ell| \leq \sqrt{\epsilon_2} = 2^{-\Omega(\ell)}$. Let

$$Bad_S = \{y \in \mathsf{Supp}(Y) : \forall i \in S, |W_i - U_\ell| \leq \sqrt{\epsilon_2} \text{ but } |(Z^i, i \in S) - U_{h\ell}| > \epsilon'\}$$

for some $\epsilon' = 2^{-\Omega(\ell)}$. Then we must have $y \in Bad_S$. Therefore, any bad y must be in $\bigcup_S Bad_S$. By the union bound we know

$$\Pr_{y \leftarrow Y}[y \text{ is bad}] \leq \sum_S \Pr[Bad_S].$$

Thus to bound the probability of a bad y we only need to bound $\Pr[Bad_S]$.

Now fix any subset $S \subseteq [N]$ with $|S| = h$. Let $R = \{\mathsf{Ext}_1(Y, r_i), i \in S\}$. We now bound $\Pr[Bad_S]$ as follows. Define

$$A_S = \{r \in \mathsf{Supp}(R) : \forall i \in S, |W_i - U_\ell| \leq \sqrt{\epsilon_2}\}.$$

Then

$$\Pr[Bad_S] = \sum_{r \in A_S} \Pr[R = r] \Pr[Bad_S | R = r].$$

We now estimate $\Pr[Bad_S | R = r]$. First we know that conditioned on any $R = r$, we have that $\forall i \in S, |W_i - U_\ell| \leq \sqrt{\epsilon_2}$. Thus by Lemma III.8 we can get rid of the error one by one for each $i \in S$ and we have that there exists another random variable $(W'_i, i \in S)$ such that $\forall i \in S$, $W'_i = U_\ell$ and $|(W_i, i \in S) - (W'_i, i \in S)| \leq h\sqrt{\epsilon_2}$. From now on we'll think of $(W_i, i \in S)$ as being $(W'_i, i \in S)$ (i.e., every row is truly uniform). This only adds $h\sqrt{\epsilon_2}$ to the final error. Now, since the size of R is bounded by $h\ell$, by Lemma III.7 we have that

$$\Pr_{r \leftarrow R}[H_\infty(Y|R = r) \geq 2k - h\ell - \ell \geq k] \geq 1 - 2^{-\ell}.$$

Now we have the following two cases.

Case 1: $H_\infty(Y|R = r) < k$. In this case we'll just bound $\Pr[Bad_S | R = r]$ by $\Pr[Bad_S | R = r] \leq 1$. However, the probability of such $R = r$ is at most $2^{-\ell}$.

Case 2: $H_\infty(Y|R = r) \geq k$. In this case, we know that $\forall i \in S$, W_i is uniform and independent of Y (since it is a deterministic function of X conditioned on the fixing of $R = r$). Thus by Theorem III.4 we have that

$$|(Y^i, W_i) - (U_{0.9k}, W_i)| \leq \epsilon_2.$$

Therefore $\forall i \in S$, we have that with probability $1 - \sqrt{\epsilon_2}$ over the fixing of W_i, Y^i is $\sqrt{\epsilon_2}$-close to uniform. Let $W = \{W_i, i \in S\}$. Then with probability $1 - h\sqrt{\epsilon_2}$ over the fixing of W, we have that each Y^i is $\sqrt{\epsilon_2}$-close to uniform. Thus again by Lemma III.8, we have that $Y^S = \{Y^i, i \in S\}$ is $h\sqrt{\epsilon_2}$-close to another source $Y'^S = \{Y'^i, i \in S\}$ where $\forall i, Y'^i = U_{0.9k}$. Now since the size of W is $h\ell$, again by Lemma III.7 we have that with probability $1 - 2^{-\ell}$ over the fixing of W, X still has min-entropy at least k. Thus, in summary, with probability $1 - h\sqrt{\epsilon_2} - 2^{-\ell}$ over the fixing of W, we have that X has min-entropy at least k, $Y^S = \{Y^i, i \in S\}$ is $h\sqrt{\epsilon_2}$-close to $Y'^S = \{Y'^i, i \in S\}$, and X and Y^S are independent (since W is a deterministic function of X). Assume for now that Y^S is just Y'^S, then we can apply Lemma V.5 to conclude that in this case, we have

$$|((Z^i, i \in S), Y) - (U_{h\ell}, Y)| \leq O(bh^2\epsilon_2).$$

Therefore with probability $1 - O(bh\sqrt{\epsilon_2})$ over the fixing of Y, we have that $|(Z^i, i \in S) - U_{h\ell}| \leq h\sqrt{\epsilon_2}$. Now adding back all the errors, we get that with probability $1 - O(bh\sqrt{\epsilon_2}) - h\sqrt{\epsilon_2} = 1 - O(bh\sqrt{\epsilon_2})$ over the fixing of Y, we have that

$$|(Z^i, i \in S) - U_{h\ell}| \leq h\sqrt{\epsilon_2} + h\sqrt{\epsilon_2} + h\sqrt{\epsilon_2} + 2^{-\ell} \leq (3h+1)\sqrt{\epsilon_2}.$$

Now let $\epsilon' = (3h+1)\sqrt{\epsilon_2} = 2^{-\Omega(\ell)}$ since $\epsilon_2 = 2^{-\Omega(\ell)}$, $\ell = k^\beta$ and $h < 2k^\alpha$. We have that in Case 2,

$$\Pr[Bad_S | R = r] \leq O(bh\sqrt{\epsilon_2}).$$

Therefore for any fixed S, we have that

$$\Pr[Bad_S] \leq 2^{-\ell} + \Pr[A_S] O(bh\sqrt{\epsilon_2}) = O(bh\sqrt{\epsilon_2}) = 2^{-\Omega(\ell)},$$

since $b < \log n = k^{O(1)}$ and $h < 2k^\alpha$.

Thus

$$\Pr_{y \leftarrow Y}[y \text{ is bad}] \leq \binom{N}{h} 2^{-\Omega(\ell)} < N^h 2^{-\Omega(\ell)} = 2^{-\Omega(\ell) + O(h \log n)} = 2^{-\Omega(\ell)},$$

if we choose h, ℓ such that $\ell \geq Ch \log n$ for some sufficiently large constant $C > 1$.

Now subtracting the probability mass of the bad y's, we get that with probability $1 - 2^{-k} - 2^{-\Omega(\ell)} = 1 - 2^{-\Omega(\ell)}$ over the fixing of Y, there exists a subset $T \subseteq [N]$ such that $|T| \geq \frac{2}{3}N$ and $\forall S \subseteq T$ with $|S| = h$, we have

$$|(Z_i, i \in S) - U_{h\ell}| \leq \epsilon' = 2^{-\Omega(\ell)}.$$

■

Next we describe the lightest bin protocol, defined in [19].

Lightest bin protocol: Assume there are N strings $\{z^i, i \in [N]\}$ where each $z_i \in \{0,1\}^m$ with $m > \log N$. The output of a lightest bin protocol with $r < N$ bins is a subset $T \subset [N]$ that is obtained as follows. Imagine that each string z^i is associated with a player P_i. Now, for each i, P_i uses the first $\log r$ bits of z_i to select a bin j, i.e., if the first $\log r$ bits of z_i is the binary expression of $j-1$, then P_i selects bin j. Now let bin l be the bin that is selected by the fewest number of players. Then

$$T = \{i \in [N] : P_i \text{ selects bin } l.\}$$

The following lemma is proved in [2].

Lemma V.8. *For every constant $0 < \gamma < 1$ there exists a constant $C_1 > 1$ such that the following holds. For any $n, k, m, N \in \mathbb{N}$, any even integer $h \geq C_1$ and any $\epsilon > 0$ with $N \geq h^2$, $\epsilon < N^{-6h}$, $k > 20h(\log n + \log(1/\epsilon))$ and $m > 10(\log n + \log(1/\epsilon))$,[2] assume that we have N sources $\{Z_1^i, i \in [N]\}$ over m bits and a subset $S \subset [N]$ with $|S| \geq \delta N$ for some constant $\delta > 1/2$, such that for any $S' \subset S$ with $|S'| = h$, we have*

[2]The constants actually depend on the hidden constant in the seed length $d = O(\log(n/\epsilon))$ of an optimal seeded extractor. Nevertheless they are always constants and don't really affect our analysis. For simplicity and clarity we use 20, 10 here.

$$(Z_1^i, i \in S') \approx_\epsilon U_{hm}.$$

Let $Z_1 = Z_1^1 \circ \cdots \circ Z_1^N$. Use Z_1 to run the lightest bin protocol with $r = \frac{\gamma^2}{16h} N^{1-\frac{2}{\sqrt{h}}}$ bins [3] and let the output contain N_2 elements $\{i_1, i_2, \cdots, i_{N_2} \in [N]\}$. Assume that X is an (n,k) source independent of Z_1. For any $j \in [N_2]$, let $Z_2^j = \mathsf{Ext}(X, Z_1^{i_j})$ where Ext is the strong seeded extractor in theorem III.4 that has seed length m and outputs $m_2 = k/(2h)$ bits with error ϵ. Then with probability at least $1 - N^{-\sqrt{h}/2}$ over the fixing of Z_1, there exists a subset $S_2 \subset [N_2]$ with $|S_2| \geq \delta(1-\gamma)N/r \geq \delta(1-\gamma)N_2$ such that for any $S_2' \subset S_2$ with $|S_2'| = h$, we have

$$(Z_2^i, i \in S_2') \approx_{\epsilon_2} U_{hm_2}$$

with $\epsilon_2 < N_2^{-6h}$ and $m_2 > 10(\log n + \log(1/\epsilon_2))$.

We can now present our construction of extractors for independent sources.

Algorithm V.9 (Independent Source Extractor IExt)**.**

Input: X — an $(n, 2k)$-source with $k \geq \frac{1}{2}\log^{12} n$. $Y = (Y_1, Y_2)$ — a $(2k, 2k)$ block source where each block has n bits, independent of X.
Output: V — a random variable close to uniform.

Sub-Routines and Parameters:
Let SR be the function in Algorithm V.6. Let $\mathsf{BasicExt}$ be the extractor in Theorem III.5. Let Ext be the strong extractor in Theorem III.4. Let $0 < \alpha < \beta < 1$ be the two constants defined before. Let $0 < \gamma < 1$ be the constant in Lemma V.8. We will choose $\alpha = 1/6, \beta = 1/3$ and $\gamma = 1/4$. Let h, ℓ be the two parameters in Algorithm V.3 with $k^\alpha \leq h < 2k^\alpha$ and $\ell = k^\beta$.

1) Compute $Z = Z^1 \circ \cdots \circ Z^N = \mathsf{SR}(X, Y_1)$.
2) Let $N = \mathrm{poly}(n)$ be the number of rows in Z. Run the lightest bin protocol with Z and $r = \frac{\gamma^2}{16h} N^{1-\frac{2}{\sqrt{h}}}$ bins and let the output contain N_1 elements $\{i_1, i_2, \cdots, i_{N_1} \in [N]\}$. Let $Z_1 = Z_1^1 \circ \cdots \circ Z_1^{N_1}$ be the concatenation of the corresponding rows in Z (i.e., $Z_1^j = Z^{i_j}$).
3) Note that $N_1 \leq \lfloor N/r \rfloor$. Without loss of generality assume that $N_1 = \lfloor N/r \rfloor$. If not, add rows of all 0 strings to Z_1 until $N_1 = \lfloor N/r \rfloor$.
4) For any $j \in [N_1]$, compute $Z_2^j = \mathsf{Ext}(Y_2, Z_1^j)$ and output $m_2 = \sqrt{k}$ bits. Let $Z_2 = Z_2^1 \circ \cdots \circ Z_2^{N_1}$.
5) For any $j \in [N_1]$, compute $Z_3^j = \mathsf{Ext}(X, Z_2^j)$ and output $m_3 = 1.9k$ bits. Let $Z_3 = Z_3^1 \circ \cdots \circ Z_3^{N_1}$.
6) Compute $V = \mathsf{BasicExt}(Y_2, Z_3)$.

We now have the following theorem.

Theorem V.10. *There exists a constant $C_0 > 1$ such that for any $n, k \in \mathbb{N}$ with $n \geq C_0$ and $k \geq \frac{1}{2}\log^{12} n$, if X is an $(n, 2k)$-source and $Y = (Y_1, Y_2)$ is an independent $(2k, 2k)$ block source where each block has n bits, then*

$$|(\mathsf{IExt}(X,Y), Y) - (U_m, Y)| \leq \epsilon$$

and

$$|(\mathsf{IExt}(X,Y), X) - (U_m, X)| \leq \epsilon,$$

where $m = 1.8k$ and $\epsilon = 2^{-k^{\Omega(1)}}$.

[3] For simplicity, we assume that r is a power of 2. If not, we can always replace it with a power of 2 that is at most $2r$. This does not affect our analysis.

Proof: By Lemma V.7, with probability $1 - 2^{-\Omega(\ell)}$ over the fixing of Y_1, there exists a subset $T \subseteq [N]$ such that $|T| \geq \frac{2}{3}N$ and $\forall S \subseteq T$ with $|S| = h$, we have

$$|(Z^i, i \in S) - U_{h\ell}| \leq 2^{-\Omega(\ell)}.$$

We now want to apply Lemma V.8. But first let's check that the conditions of Lemma V.7 and Lemma V.8 are satisfied. Note that $k^\alpha \leq h < 2k^\alpha$, $\ell = k^\beta$ and $b < \log n$. To apply Lemma V.7, we need that $k \geq 2(bh+2)(h^2+12)\ell$ and $\ell \geq Ch \log n$ for some sufficiently large constant $C > 1$. To apply Lemma V.8, we need that $\epsilon' < N^{-6h}$, $k > 20h(\log n + \log(1/\epsilon'))$ and $m = \ell > 10(\log n + \log(1/\epsilon'))$. In Algorithm V.6 we also need $k \geq (h+1)\ell$. Altogether, it suffices to have $0 < \alpha < \beta < 1$ satisfy the following conditions.

$$k \geq 3 \log n h^3 \ell,\ \ell \geq Ch \log n, \epsilon' < N^{-6h} \text{ and } \ell > 10(\log n + \log(1/\epsilon')).$$

These conditions are satisfied if the following conditions are satisfied.

$$k \geq 24k^{3\alpha+\beta} \log n \text{ and } \ell = k^\beta \geq Ck^\alpha \log n$$

for some constant $C > 1$.

Now if $\alpha = 1/6, \beta = 1/3$ and $k \geq \frac{1}{2}\log^{12} n$, then we see that for sufficiently large n,

$$\frac{k}{k^{3\alpha+\beta}} = k^{1/6} \geq \Omega(\log^2 n) > 24 \log n \text{ and } \frac{k^\beta}{k^\alpha} = k^{1/6} \geq \Omega(\log^2 n) > C \log n.$$

Thus the above conditions are satisfied.

Notice that $m_2 = \sqrt{k} < k/(2h)$, thus by Lemma V.8 we have that with probability at least $1 - N^{-\sqrt{h}/2}$ over the fixing of Z, there exists a subset $S \subset [N_1]$ with $|S| \geq \delta(1-\gamma)N/r \geq \frac{2}{3}\frac{3}{4}N/r = \frac{1}{2}N/r$ such that for any $S' \subset S$ with $|S'| = h$, we have

$$(Z_2^i, i \in S') \approx_{\epsilon_2} U_{h\sqrt{k}}$$

with $\epsilon_2 < N_1^{-6h}$.

Note that Z_2 is a deterministic function of Y_2 and Z_1, and Z_1 is a deterministic function of Z. Thus we also have that with probability at least $1 - N^{-\sqrt{h}/2}$ over the fixing of Z_1, the above property holds. Also note that $N/r = \frac{16h}{\gamma^2}N^{\frac{2}{\sqrt{h}}} > 16h$, so $|S| > 8h > 1$. Thus with probability at least $1 - N^{-\sqrt{h}/2}$ over the fixing of Z_1, we have that Z_2 is $N_1^{-6h} < (8h)^{-6h}$-close to an SR source (since $N_1 \geq |S|$).

Note that conditioned on the fixing of Z_1, we have that Z_2 is a deterministic function of Y_2, and is thus independent of X. Now note that $N/r = \frac{16h}{\gamma^2}N^{\frac{2}{\sqrt{h}}}$. Since $h \geq k^\alpha = k^{1/6}$ and $k \geq \frac{1}{2}\log^{12} n$, we have that

$$N^{\frac{2}{\sqrt{h}}} \leq \text{poly}(n)^{O(1/\log n)} = O(1).$$

Thus $N_1 \leq N/r = O(h) < k^{1/4}$. Note that conditioned on the fixing of Y_1, we have that Z_1 is a deterministic function of X, with the size of Z_1 bounded by $k^{1/4}\ell < k^{2/3}$. Therefore by Lemma III.7, we have that with probability $1 - 2^{-0.05k}$ over the fixing of Z_1, X still has min-entropy at least $2k - k^{2/3} - 0.05k > 1.94k$.

Now since Z_2 is independent of X and assuming that Z_2 is indeed an SR-source, then by Theorem III.4 we have that for some $i \in [N_1]$,

$$|(Z_3^i, Z_2^i) - (U_{1.9k}, Z_2^i)| \leq 2^{-\Omega(\sqrt{k})}.$$

Thus with probability at least $1 - 2^{-\Omega(\sqrt{k})}$ over the fixing of Z_2^i (and thus also the fixing of Z_2), we have that Z_3 is $2^{-\Omega(\sqrt{k})}$-close to an $N_1 \times 1.9k$ SR-source. Moreover, conditioned on the further fixing of Z_2, we have that Z_3 is a deterministic function of X, and is thus independent of Y_2. Furthermore, note the size of Z_2 is bounded by $N_1\sqrt{k} \leq k^{1/4}\sqrt{k} = k^{3/4}$. Thus again by Lemma III.7, we have that with probability $1 - 2^{-0.05k}$ over the fixing of Z_2, Y_2 still has min-entropy at least $2k - k^{3/4} - 0.05k > 1.94k$.

Note that $N_1 < k^{1/4}$ and $k^{1-2/4} = k^{1/2} > \log^{1.1} n$, thus by Theorem III.5, we have that

$$|(V, Y_2) - (U_m, Y_2)| \leq \epsilon_2$$

and

$$|(V, Z_3) - (U_m, Z_3)| \leq \epsilon_2,$$

where $m = 1.8k$ and $\epsilon_2 = 2^{-k^{\Omega(1)}}$. Since we have already fixed Y_1, Z_1 and Z_2, we have that Z_3 is a deterministic function of X. Thus conditioned on Z_3, we have that V is a deterministic function of Y_2, which is independent of X. Thus we also have that

$$|(V, X) - (U_m, X)| \leq \epsilon$$

and

$$|(V, Y) - (U_m, Y)| \leq \epsilon,$$

where by adding back all the errors we have

$$\epsilon \leq \epsilon_2 + 2^{-\Omega(\ell)} + N^{-\sqrt{h}/2} + (8h)^{-6h} + 2^{-0.05k} + 2^{-\Omega(\sqrt{k})} + 2^{-\Omega(\sqrt{k})} + 2^{-0.05k} = 2^{-k^{\Omega(1)}}.$$[4]

■

Note that when $n < C_0$, the extractor can be constructed in constant time just by exhaustive search (in fact, we can get a two-source extractor in this way). Thus, we have the following theorem (by replacing $2k$ with k).

Theorem V.11. *For all $n, k \in \mathbb{N}$ with $k \geq \log^{12} n$, there is an efficiently computable function $\mathsf{IExt} : \{0,1\}^n \times \{0,1\}^{2n} \to \{0,1\}^m$ such that if X is an (n,k)-source and $Y = (Y_1, Y_2)$ is an independent (k,k) block source where each block has n bits, then*

$$|(\mathsf{IExt}(X, Y), Y) - (U_m, Y)| \leq \epsilon$$

and

$$|(\mathsf{IExt}(X, Y), X) - (U_m, X)| \leq \epsilon,$$

where $m = 0.9k$ and $\epsilon = 2^{-k^{\Omega(1)}}$. [5]

As a corollary, we immediately obtain the following theorem.

Theorem V.12. *For all $n, k \in \mathbb{N}$ with $k \geq \log^{12} n$, there is an efficiently computable three-source extractor $\mathsf{IExt} : (\{0,1\}^n)^3 \to \{0,1\}^m$ such that if X, Y, Z are three independent (n,k)-sources, then*

$$|\mathsf{IExt}(X, Y, Z) - U_m| \leq \epsilon,$$

where $m = 0.9k$ and $\epsilon = 2^{-k^{\Omega(1)}}$.

If the entropy k gets very close to $\log^2 n$, then we can use a similar construction as the extractor in [2], except replacing the step of creating the initial SR-source with the method in this paper. In this case we can get an extractor for two independent block sources each with a constant number of blocks of min-entropy k. The detailed algorithm and proof are omitted here, but can be found in the full version [32]. We have the following theorem.

Theorem V.13. *For every constant $\eta > 0$ and all $n, k \in \mathbb{N}$ with $k \geq \log^{2+\eta} n$, there is an efficiently computable extractor $\mathsf{BExt} : (\{0,1\}^n)^t \times (\{0,1\}^n)^t \to \{0,1\}^m$ with $t = \lceil \frac{7}{\eta} \rceil + 1$, such that if $X = (X_1, X_2, \cdots X_t), Y = (Y_1, Y_2, \cdots Y_t)$ are two independent $(k, k, \cdots, k)$- block sources where each block has n bits, then*

[4] One can show that in this case ϵ_2 is $n^{-\omega(1)}$, as well as all the other terms. So the entire error is $n^{-\omega(1)}$.

[5] The constant 0.9 can be replaced by any constant less than 1.

$$|(\mathsf{BExt}(X,Y),Y)-(U_m,Y)|\le\epsilon$$

and

$$|(\mathsf{BExt}(X,Y),X)-(U_m,X)|\le\epsilon,$$

where $m=0.9k$ *and* $\epsilon=2^{-k^{\Omega(1)}}$.

As a corollary, we immediately obtain the following theorem.

Theorem V.14. *For every constant* $\eta>0$ *and all* $n,k\in\mathbb{N}$ *with* $k\ge\log^{2+\eta}n$*, there is an efficiently computable extractor* $\mathsf{IExt}:(\{0,1\}^n)^t\to\{0,1\}^m$ *with* $t=\lceil\frac{14}{\eta}\rceil+2$ *such that if* $X_1,\cdots,X_t$ *are* t *independent* (n,k)*-sources, then*

$$|\mathsf{IExt}(X_1,\cdots,X_t)-U_m|\le\epsilon,$$

where $m=0.9k$ *and* $\epsilon=2^{-k^{\Omega(1)}}$.

ACKNOWLEDGMENT

We thank the anonymous reviewers of FOCS 2015 for helpful comments.

REFERENCES

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

[2] X. Li, "Extractors for a constant number of independent sources with polylogarithmic min-entropy," in *Proceedings of the 54th Annual IEEE Symposium on Foundations of Computer Science*, 2013, pp. 100–109.

[3] N. Nisan and D. Zuckerman, "Randomness is linear in space," *Journal of Computer and System Sciences*, vol. 52, no. 1, pp. 43–52, 1996.

[4] C. J. Lu, O. Reingold, S. Vadhan, and A. Wigderson, "Extractors: Optimal up to constant factors," in *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, 2003, pp. 602–611.

[5] V. Guruswami, C. Umans, and S. Vadhan, "Unbalanced expanders and randomness extractors from Parvaresh-Vardy codes," *Journal of the ACM*, vol. 56, no. 4, 2009.

[6] Z. Dvir and A. Wigderson, "Kakeya sets, new mergers and old extractors," in *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, 2008.

[7] Z. Dvir, S. Kopparty, S. Saraf, and M. Sudan, "Extensions to the method of multiplicities, with applications to kakeya sets and mergers," in *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science*, 2009.

[8] Y. T. Kalai, X. Li, A. Rao, and D. Zuckerman, "Network extractor protocols," in *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, 2008, pp. 654–663.

[9] Y. Kalai, X. Li, and A. Rao, "2-source extractors under computational assumptions and cryptography with defective randomness," in *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science*, 2009, pp. 617–628.

[10] B. Chor and O. Goldreich, "Unbiased bits from sources of weak randomness and probabilistic communication complexity," *SIAM Journal on Computing*, vol. 17, no. 2, pp. 230–261, 1988.

[11] B. Barak, A. Rao, R. Shaltiel, and A. 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.

[12] X. Li, "A new approach to affine extractors and dispersers," in *Proceedings of the 26th Annual IEEE Conference on Computational Complexity*, 2011, pp. 137–147.

[13] J. Kamp, A. Rao, S. Vadhan, and D. Zuckerman, "Deterministic extractors for small space sources," in *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006.

[14] B. Barak, R. Impagliazzo, and A. Wigderson, "Extracting randomness using few independent sources," in *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, 2004, pp. 384–393.

[15] B. Barak, G. Kindler, R. Shaltiel, B. Sudakov, and A. Wigderson, "Simulating independence: New constructions of condensers, Ramsey graphs, dispersers, and extractors," in *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, 2005, pp. 1–10.

[16] R. Raz, "Extractors with weak random seeds," in *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, 2005, pp. 11–20.

[17] A. 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.

[18] X. Li, "Improved constructions of three source extractors," in *Proceedings of the 26th Annual IEEE Conference on Computational Complexity*, 2011, pp. 126–136.

[19] ——, "New independent source extractors with exponential improvement," in *Proceedings of the 45th Annual ACM Symposium on Theory of Computing*, 2013, pp. 783–792.

[20] J. Radhakrishnan and A. Ta-Shma, "Bounds for dispersers, extractors and depth-two superconcentrators," *Siam Journal on Discrete Mathematics*, vol. 13, pp. 2–24, 2000.

[21] U. Feige, "Noncryptographic selection protocols," in *Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science*, IEEE, Ed. IEEE Computer Society Press, 1999, pp. 142–152.

[22] D. Zuckerman, "Randomness-optimal oblivious sampling," *Random Structures and Algorithms*, vol. 11, pp. 345–367, 1997.

[23] S. Dziembowski and K. Pietrzak, "Leakage-resilient cryptography," in *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, 2008.

[24] Y. Dodis and D. Wichs, "Non-malleable extractors and symmetric key cryptography from weak secrets," in *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, 2009, pp. 601–610.

[25] X. Li, "Non-malleable extractors, two-source extractors and privacy amplification," in *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science*, 2012, pp. 688–697.

[26] ——, "Non-malleable condensers for arbitrary min-entropy, and almost optimal protocols for privacy amplification," in *12th IACR Theory of Cryptography Conference*. Springer-Verlag, 2015, pp. 502–531, lNCS 9014.

[27] G. Cohen, "Local correlation breakers and applications to three-source extractors and mergers," in *Proceedings of the 56th Annual IEEE Symposium on Foundations of Computer Science*, 2015.

[28] ——, "Two-source dispersers for polylogarithmic entropy and improved ramsey graphs," ECCC, Tech. Rep. TR15-095, 2015.

[29] E. Chattopadhyay and D. Zuckerman, "Explicit two-source extractors and resilient functions," Electronic Colloquium on Computational Complexity, Tech. Rep. TR15-119, 2015.

[30] X. Li, "Improved constructions of two-source extractors," Electronic Colloquium on Computational Complexity, Tech. Rep. TR15-125, 2015.

[31] U. M. Maurer and S. Wolf, "Privacy amplification secure against active adversaries," in *Advances in Cryptology — CRYPTO '97, 17th Annual International Cryptology Conference, Proceedings*, 1997.

[32] X. Li, "Three source extractors for polylogarithmic min-entropy," Electronic Colloquium on Computational Complexity, Tech. Rep. TR15-034, 2015.