2-Source Dispersers for $n^{o(1)}$ Entropy, and Ramsey Graphs Beating the Frankl-Wilson Construction

Boaz Barak* Anup Rao† Ronen Shaltiel‡ Avi Wigderson§

July 22, 2008

Abstract

The main result of this paper is an explicit disperser for two independent sources on n bits, each of min-entropy $k = 2^{\log^{1-\alpha_0} n}$, for some small absolute constant $\alpha_0 > 0$). Put differently, setting $N = 2^n$ and $K = 2^k$, we construct an explicit $N \times N$ Boolean matrix for which no $K \times K$ sub-matrix is monochromatic. Viewed as the adjacency matrix of a bipartite graph, this gives an explicit construction of a *bipartite K-Ramsey* graph of $2N$ vertices.

This improves the previous the previous bound of $k = o(n)$ of Barak, Kindler, Shaltiel, Sudakov and Wigderson [BKS+05]. As a corollary, we get a construction of a $2^{2^{\log^{1-\alpha_0} n}}$ (non bipartite) Ramsey graph of 2^n vertices, significantly improving the previous bound of $2^{\tilde{O}(\sqrt{n})}$ due to Frankl and Wilson [FW81].

We also give a construction of a new independent sources extractor that can extract from a constant number of sources of polynomially small min-entropy with exponentially small error. This improves independent sources extractor of Rao [Rao06], which only achieved polynomially small error.

Our dispersers combine ideas and constructions from several previous works in the area together with some new ideas. In particular, we rely on the extractors of Raz [Raz05] and Bourgain [Bou05] as well as an improved version of the extractor of Rao [Rao06]. A key ingredient that allows us to beat the barrier of $k = \sqrt{n}$ is a new and more complicated variant of the *challenge-response mechanism* of Barak et al. [BKS+05] that allows us to locate the min-entropy concentrations in a source of *low* min-entropy.

Keywords: Extractors, Dispersers, Ramsey Graphs

*Department of Computer Science, Princeton University, boaz@cs.princeton.edu. Supported by United States-Israel Binational Foundation (BSF) grant .

†Institute for Advanced Study, Princeton, New Jersey, arao@ias.edu. Much of this work was done while the author was a student at the University of Texas, visiting Princeton University and the Institute for Advanced Study. Supported in part by an MCD fellowship from UT Austin and NSF Grant CCR-0310960.

‡Ronen Shaltiel, University of Haifa, Mount Carmel, Haifa, Israel, ronen@cs.haifa.ac.il. This research was supported by the United States-Israel Binational Science Foundation (BSF) grant 2004329.

§Institute for Advanced Study, Princeton, New Jersey, avi@math.ias.edu. Supported by NSF grant CCR 0324906.

Contents

1 Introduction

In this paper we give new explicit constructions of certain combinatorial objects. The results can be described in two equivalent ways. The first, which is simpler and has a longer history, is the language of *Ramsey graphs*, graphs that do not have large cliques or independent sets. The second is the language of *randomness extractors* and *randomness dispersers*. While a bit more complicated to state, this latter form is key to both the Computer Science motivation of the problem, and our actual techniques.

1.1 Ramsey and Bipartite Ramsey Graphs.

We start by describing our results in the language of Ramsey graphs:

Definition 1.1. A graph on N vertices is called a *K-Ramsey Graph* if it contains no clique or independent set of size K.

In 1928 Ramsey [Ram28] proved that there does *not* exist a graph on $N = 2^n$ vertices that is $n/2$ Ramsey. In 1947 Erdős published his paper inaugurating the *Probabilistic Method* with a few examples, including a proof that complemented Ramsey's discovery: *most* graphs on 2^n vertices are $2n$-Ramsey. The quest for constructing Ramsey graphs *explicitly* has existed ever since and led to some beautiful mathematics. By an *explicit construction* we mean an efficient (i.e., polynomial time) algorithm that, given the labels of two vertices in the graph, determines whether there is an edge between them.[1]

Prior to this work, the best record was obtained in 1981 by Frankl and Wilson [FW81], who used intersection theorems for set systems to construct N-vertex graphs that are $2^{\tilde{\Omega}(\sqrt{n})}$-Ramsey [2]. This bound was matched by Alon [Alo98] using the *Polynomial Method*, by Grolmusz [Gro00] using low rank matrices over rings, and also by Barak [Bar06] boosting Abbot's method with almost k-wise independent random variables (a construction that was independently discovered by others as well). Remarkably all of these different approaches got stuck at essentially the same bound. In recent work, Gopalan [Gop06] showed that other than the last construction, all of these can be viewed as coming from low-degree symmetric representations of the OR function. He also shows that any such symmetric representation cannot be used to give a better Ramsey graph, suggesting why these constructions achieved such similar bounds. Indeed, as we will discuss in a later section, the $\sqrt{n}$ min-entropy bound initially looked like a natural obstacle even for our techniques, though eventually we were able to surpass it.

One can make an analogous definition for *bipartite* graphs:

Definition 1.2. A bipartite graph on two sets of N vertices is a bipartite *K-Ramsey Graph* if it has no $K \times K$ complete or empty bipartite subgraph.

Given a bipartite K-Ramsey graph G on $2N$ vertices, one can easily transform it into a non-bipartite $K/2$-Ramsey graph H on N vertices[3]. Thus, the problem of explicitly constructing bipartite Ramsey graphs is at least as hard as the problem of constructing non-bipartite Ramsey graphs. Indeed,

[1] Almost all of the constructions mentioned below (including our own) achieve this definition, with the exception of the papers [Bar06, PR04] that achieve a somewhat weaker notion of explicitness.

[2] We use $\tilde{O}$ and $\tilde{\Omega}$ notations when neglecting polylogarithmic factors.

[3] The $N \times N$ adjacency matrix of a bipartite Ramsey graph is not necessarily symmetric and may contain ones on the diagonal. This can be fixed by using only the upper triangle of the matrix (e.g., by placing an edge $\{a, b\}$ in H, where $a < b$, if the a^{th} vertex on the left side is connected to the b^{th} vertex on the right side in G). It is easy to verify that this indeed yields a $K/2$-Ramsey graph.

while Erdős' result on the abundance of $2n$-Ramsey graphs holds as is for bipartite graphs, the best explicit construction of bipartite Ramsey graphs only recently surpassed the bound of $2^{n/2}$ that is given by the Hadamard matrix. The bound was first improved to $o(2^{n/2})$ by Pudlak and Rődl [PR04] and then to $2^{o(n)}$ by Barak, Kindler, Shaltiel, Sudakov and Wigderson [BKS+05].

The main result of this paper is a new bound that improves the state of affairs for both the bipartite and non-bipartite cases.

Theorem 1.3 (Main Theorem)**.** *There is an absolute constant $\alpha_0 > 0$ and an explicit construction of a bipartite $2^{2^{\log^{1-\alpha_0} n}} = 2^{n^{o(1)}}$-Ramsey graph over $2 \cdot 2^n$ vertices, for every large enough $n \in \mathbb{N}$.*

As discussed above, this corollary follows easily:

Corollary 1.4. *There is an absolute constant $\alpha_0 > 0$ and an explicit construction of a $2^{2^{\log^{1-\alpha_0} n}} = 2^{n^{o(1)}}$ Ramsey graph over 2^n vertices, for every large enough $n \in \mathbb{N}$.*

1.2 Randomness extractors.

We now describe our results in a different language — the language of *randomness extractors* and *randomness dispersers*. We start with some background. The use of randomness in Computer Science has gained tremendous importance in the last few decades. Randomness now plays an important role in algorithms, distributed computation, cryptography and many more areas. Some of these applications have been shown to *inherently require* a source of randomness. However, it is far from clear where the randomness that is needed for these applications can be obtained.

An obvious approach is to use a natural source of *unpredictable data* such as users' typing rates, radioactive decay patterns, fluctuations in the stock market, etc. However, when designing randomized algorithms and protocols, it is almost always assumed that a sequence of *unbiased* and *independent* coin tosses is available, while natural unpredictable data do not necessarily come in that form.

One way to attempt to close this gap is to apply some kind of hash function that is supposed to transform the unpredictable/high entropy data into a distribution that is equal (or at least close to) the uniform distribution. To formalize this approach, let us model *weak sources* as probability distributions over n bit strings that have sufficient min-entropy[4] k. Such a source is referred to as a k-source. One then seeks a function f, called an *extractor*, that maps $\{0,1\}^n$ to $\{0,1\}^m$ (for m as large as is feasible) such that for every random variable X with sufficient min-entropy, $f(X)$ is close to the uniform distribution. Unfortunately this goal can never be met: it is impossible even if we want to output just a single bit from distributions of very high min-entropy (i.e., random variables over $\{0,1\}^n$ with min-entropy $n-1$), as for every function $f : \{0,1\}^n \to \{0,1\}$, we can find $2^n/2$ inputs on which f is constant. Thus, the uniform distribution over these inputs is a distribution on which f fails to extract randomness.

One way to break out of this conundrum, suggested by Santha and Vazirani [SV86] and Chor and Goldreich [CG88], is to use *more than one input source of randomness.* That is, we consider the case that the extractor function gets samples from *several independent sources of randomness.* The probabilistic method can be used to show that in principle two independent sources suffice — for every

[4] It turns out that min-entropy, and not Shannon entropy, is the right notion of entropy to use in this context. A random variable X has min entropy at least k if for every x in X's range $\Pr[X = x] \leq 2^{-k}$. A special case is *flat distributions* (these are distributions that are uniformly distributed over some subset of $\{0,1\}^n$ which is of size 2^k). Note that for flat distributions, entropy and min-entropy coincide. Furthermore, any distribution with min-entropy k is a convex combination of such flat distributions and therefore the reader can without loss of generality assume that all sources of randomness are flat distributions.

$n \in \mathbb{N}$, $\epsilon > 0$ and $k \geq 2\log n + 10\log 1/\epsilon$ there exists a function $f : (\{0,1\}^n)^2 \to \{0,1\}^{0.9k}$ such that for every two independent distributions X, Y each having min-entropy at least k, $f(X,Y)$ is within ϵ statistical distance[5] to the uniform distribution over $\{0,1\}^{0.9k}$. Such a function f is called a *two source extractor*. Formally, we make the following definition:

Definition 1.5. Let $n, c, k \in \mathbb{N}$ and $\epsilon > 0$. A function $f : (\{0,1\}^n)^c \to \{0,1\}^m$ is called a *c-source extractor* for min-entropy k with error ϵ, if for every independent random variables $X_1, \ldots, X_c$ each having min-entropy at least k,

$$|f(X_1, \ldots, X_c) - U_m| < \epsilon \quad ,$$

where U_m denotes the uniform distribution over $\{0,1\}^m$.

The probabilistic method shows the existence of an excellent extractor in terms of all the parameters. However, to be useful in computer science applications, the extractor needs to be *efficiently computable*. In other words, we need an *explicit construction* that matches, or at least gets close to, the bounds achieved by the probabilistic method. Beyond the obvious motivations (potential use of physical sources for randomized computation), extractors have found applications in a variety of areas in theoretical computer science where randomness does not seem an issue, such as in efficient constructions of communication networks [WZ99, CRVW02], error correcting codes [TZ04, Gur03], data structures [MNSW98] and more. (Many of the applications and constructions are for a related notion called *seeded* extractors, which are two source extractor in which the second source is very short but assumed to be completely uniform, see [Sha02] for a survey of much of this work.)

Until a few years ago, essentially the only known explicit construction for a constant number of sources was the Hadamard extractor Had defined by $\mathsf{Had}(x,y) = \langle x, y \rangle \mod 2$. It is a two-source extractor for min-entropy $k > n/2$ as observed by Chor and Goldreich [CG88] and can be extended to give $\Omega(n)$ output bits as observed by Vazirani [Vaz85]. Roughly 20 years later, Barak, Impagliazzo and Wigderson [BIW04] constructed a $c = O(\log(n/k))$-source extractor for min-entropy k with output $m = \Omega(k)$. Note that this means that if $k = \delta n$ for a constant δ, this extractor requires only a constant number of sources. The main tool used by Barak et. al. was a breakthrough in additive number theory of Bourgain, Katz and Tao [BKT04] who proved a finite-field *sum product theorem*, a result that has already found applications in diverse areas of mathematics, including analysis, number theory, group theory and ... extractor theory. Building on these works, Barak et al. [BKS+05] and Raz [Raz05] independently gave constructions of extractors for just three sources with min-entropy $k = \delta n$ for any constant $\delta > 0$. This was followed by a result of Rao [Rao06], who showed how to extract from $O(\log n / \log k)$ independent k-sources. This is an extractor for $O(c)$ sources as long as k is larger than $n^{1/c}$. His extractor did not rely on any of the new results from additive number theory. In this paper, we extend Rao's results by improving the error parameter from $\epsilon = k^{-\Omega(1)}$ to $\epsilon = 2^{-k^{\Omega(1)}}$, a result that was also obtained independently by Chung and Vadhan [CV].

Theorem 1.6. *There is a polynomial time computable c-source extractor $f : (\{0,1\})^c \to \{0,1\}^{\Omega(k)}$ for min-entropy $k > \log^{10} n$, $c = O(\frac{\log n}{\log k})$ and $\epsilon = 2^{-k^{\Omega(1)}}$.*

Rao also gave extractors that can extract randomness from two "block-sources" with $O(\log n / \log k)$ blocks. An important ingredient in our main results is extending these results so that only one of the sources needs to be a block-source while the other source can be a general source with min-entropy k. We elaborate on block-sources and their role in our main construction later on.

[5]The *statistical distance* of two distributions W, Z over some range R, denoted by $|W - Z|$, is defined to be $1/2 \sum_{r \in R} |\Pr[W = r] - \Pr[Z = r]|$.

While the aforementioned works achieved improvements for more than two sources, the only improvement for *two source* extractors over the Hadamard extractor is by Bourgain [Bou05], who broke the "1/2 barrier" and gave such an extractor for min-entropy $.4999n$, again with linear output length $m = \Omega(n)$.

Theorem 1.7 ([Bou05])**.** *There is a polynomial time computable* 2*-source extractor* $f : (\{0,1\}^n)^2 \rightarrow \{0,1\}^m$ *for min-entropy* $.4999n$, $m = \Omega(n)$ *and* $\epsilon = 2^{-\Omega(n)}$.

This seemingly minor improvement plays an important role in Rao's extractor for two block-sources and in our improvements.

1.3 Dispersers and their relation to Ramsey graphs

A natural relaxation of extractors is, rather than requiring that the output is statistically close to the uniform distribution, simply requiring that it has large support. Such objects are called *dispersers*:

Definition 1.8. Let $n, c, k \in \mathbb{N}$ and $\epsilon > 0$. A function $f : (\{0,1\}^n)^c \rightarrow \{0,1\}^m$ is called a *c-source disperser* for min-entropy k and error parameter ϵ, if for every independent random variables $X_1, \ldots, X_c$ each having min-entropy at least k,

$$|f(X_1, \ldots, X_c)| \geq (1-\epsilon)2^m \quad .$$

We remark that in the definition above it is sufficient to consider only flat sources $X_1, \ldots, X_c$. In other words an equivalent definition is that for any c sets $S_1, \ldots, S_c \subseteq \{0,1\}^n$ such that all sets are of size 2^k, $|f(S_1 \times \ldots \times S_c)| \geq (1-\epsilon)2^m$. Dispersers are easier to construct than extractors, and in the past, progress on constructing extractors and dispersers has been often closely related.

Two source dispersers are particularly interesting as they are equivalent to bipartite Ramsey graphs. More precisely, if $f : (\{0,1\}^n)^2 \rightarrow \{0,1\}$ is a two-source disperser with one-bit output for min-entropy k and error parameter $\epsilon < 1/2$, we consider the graph G on two sets of 2^n vertices where there we place an edge from x to y if $f(x,y) = 1$. Note that any $2^k \times 2^k$ subgraph of this graph cannot be complete or empty as it must contain both an edge and a non-edge and therefore G is a bipartite 2^k-Ramsey graph. Recall that this graph can be easily transformed into a (2^{k-1})-Ramsey graph on 2^n vertices.

For 2 sources Barak et al. [BKS+05] were able to construct dispersers for sources of min-entropy $k = o(n)$:

Theorem 1.9 ([BKS+05])**.** *There exists a polynomial time computable* 2*-source disperser* $f : (\{0,1\}^n)^2 \rightarrow \{0,1\}$ *for min-entropy* $o(n)$ *and* $\epsilon < 1/2$.

The main result of this paper is a polynomial time computable disperser for two sources of min-entropy $n^{o(1)}$, improving the results of Barak et al. [BKS+05] (that achieved $o(n)$ min-entropy). By the discussion above our construction yields both bipartite Ramsey graphs and Ramsey graphs for $K = 2^{n^{o(1)}}$ and improves on Frankl and Wilson [FW81], who built Ramsey Graphs with $K = 2^{\tilde{O}(\sqrt{n})}$ (which in this terminology is a disperser for two *identically distributed* sources for min-entropy $\tilde{O}(\sqrt{n})$).

Theorem 1.10 (Main theorem, restated)**.** *There exists a constant* $\alpha_0 > 0$ *and a polynomial time computable* 2*-source disperser* $\mathsf{D} : (\{0,1\}^n)^2 \rightarrow \{0,1\}$ *for min-entropy* $2^{\log^{1-\alpha_0} n}$ *and error parameter smaller than* 1/2.

Even though our main result is a one output bit disperser, we will need to use the more general definitions of multiple-source and larger outputs dispersers and extractors in the course of our construction.

1.4 Organization of this paper

Unfortunately, our construction involves many technical details. In an attempt to make this paper more readable we also include some sections that only contain high level informal explanations, explanations that are *not* intended to be formal proofs and can be safely skipped if the reader so wishes. The paper is organized as follows:

In Section 2 we explain the high level ideas used in our construction without going into the details or giving our construction. In Section 3 we give some definitions and technical lemmas. We also state results from previous work that our work relies on. In Section 4 we present two variants of extractors that are used in our construction. The first is an extractor that extracts randomness from two independent sources where one of them is a block-source and the other is a general source. The second is a "somewhere extractor" with special properties that when given two independent sources of sufficient min-entropy outputs a polynomial number of strings where one of them is (close to) uniformly distributed. In Section 5 we give a detailed informal explanation of our construction and proof. We hope that reading this section will make it easier for the reader to follow the formal proof. In Section 6 we present our disperser construction and prove its correctness. In Section 7 we show how to construct the extractor from Section 4. Finally we conclude with some open problems.

2 Techniques

Our construction makes use of many ideas and notions developed in previous works as well as several key new ones. In this section we attempt to survey some of these at a high level without getting into precise details. In order to make this presentation more readable, we allow ourselves to be imprecise and oversimplify many issues. We stress that the contents of this section are not used in later parts of the paper. In particular, the definitions and theorems that appear in this section are restated (using precise notation) in the technical sections of the paper. The reader may skip to the more formal parts at any point if she wishes.

2.1 Subsources

Recall that the main goal of this research area is to design 2-source extractors for low min-entropy k. As explained earlier, it is unknown how to achieve extractors for two sources with min-entropy $k < 0.4999n$ and this paper only gives constructions of dispersers (rather than extractors). In order to explain how we achieve this relaxed goal, we first need the notion of subsources.

Definition 2.1 (Subsource)**.** A distribution X' over domain $\{0,1\}^n$ is a *subsource* of a distribution X (over the same domain $\{0,1\}^n$) with *deficiency* d if there exists an event $A \subseteq \{0,1\}^n$ such that $\Pr[X \in A] \geq 2^{-d}$ and X' is the probability distribution obtained by conditioning X to A. (More precisely, for every $a \in A$, $\Pr[X' = a]$ is defined to be $\Pr[X = a | X \in A]$ and for $a \notin A$, $\Pr[X' = a] = 0$).

In the case X' is a subsource of a flat distribution (a distribution that is uniform on some subset) X is simply a flat distribution on a smaller subset. It is also easy to see that if X is a k-source and X' is a deficiency d subsource of X, then X' is a $(k-d)$-source.

We say that a function $f : (\{0,1\}^n)^2 \to \{0,1\}$ is a subsource extractor if for every two independent k-sources X and Y there exist subsources X' of X and Y' of Y such that $f(X', Y')$ is close to uniformly distributed. While f is not necessarily an extractor, it certainly is a disperser, since $f(X, Y)$ is both zero and one with positive probability. Thus, when constructing dispersers, it is sufficient to analyze how our construction performs on some subsources of the adversarially chosen sources.

Our analysis uses this approach extensively. Given the initial k-sources X and Y (which can be arbitrary) we prove that there exist subsources of X, Y which have a certain "nice structure". We then proceed to design two-source extractors that extract randomness from sources with this nice structure. When using this approach, we shall be very careful to ensure that the subsources we use have low deficiency and remain a product distribution.

2.2 Block-sources

We now describe what we mean by sources that have nice structure. We consider sources that give samples which can be broken into several disjoint "blocks" such that each block has min-entropy k even conditioned on any value of the previous blocks. Called *block-sources*, these were first defined by Chor and Goldreich [CG88].

Definition 2.2 (Block-sources [CG88])**.** A distribution $X = (X_1, \ldots, X_c)$ where each X_i is of length n is a *c-block-source* of block min-entropy k if for every $i \in [c]$, every $x \in \{0,1\}^n$ and every $x_1, \ldots, x_{i-1} \in (\{0,1\}^n)^{i-1}$, $\Pr[X_i = x | X_1 = x_1 \wedge \ldots \wedge X_{i-1} = x_{i-1}] \leq 2^{-k}$.

It is clear that any such block-source is a ck-source. However, the converse is not necessarily true. Throughout this informal description, the reader should think of c as very small compared to k or n so that values like ck, k and k/c are roughly the same. Block-sources are interesting since they are fairly general (there is no deterministic way to extract from a block-source), yet we have a better understanding of how to extract from them. For example, when the input sources are block-sources with sufficiently many blocks, Rao proves that 2 independent sources suffice even for the case of lower min-entropy, with polynomially small error:

Theorem 2.3 ([Rao06])**.** *There is a polynomial time computable extractor $f : (\{0,1\}^{cn})^2 \to \{0,1\}^m$ for 2 independent c-block-sources with block min-entropy k and $m = \Omega(k)$ for $c = O((\log n)/(\log k))$.*

In this paper, we improve his result in two ways— only one of the 2 sources needs to be a c-block-source and the error is exponentially small. The other source can be an arbitrary source with sufficient min-entropy.

Theorem 2.4 (Block + General Source Extractor)**.** *There is a polynomial time computable extractor $\mathsf{B} : (\{0,1\}^n)^2 \to \{0,1\}^m$ for 2 independent sources, one of which is a c-block-source with block min-entropy k and the other a source of min-entropy k, with $m = \Omega(k)$, $c = O((\log n)/(\log k))$ and error at most $2^{-k^{\Omega(1)}}$.*

This is a central building block in our construction. This extractor, like Rao's extractor above, relies on 2-source extractor constructions of Bourgain [Bou05] and Raz [Raz05]. We do not describe how to construct the extractor of Theorem 2.4 in this informal overview. The details are in Section 4 and Section 7.

2.3 Existence of block-sources in general sources

Given that we know how to handle the case of block-sources, it is natural to try and convert a general k-source into a block-source. Let us first restrict our attention to the case where the min-entropy is high: $k = \delta n$ for some constant $\delta > 0$ (these are the parameters already achieved in the construction of [BKS+05]). We make the additional simplifying assumption that for $k = \delta n$ the extractor of Theorem 2.4 requires a block-source with only two blocks (i.e. $c = 2$).

First consider a partition of the k-source X into $t = 1/10\delta$ consecutive blocks of length n/t and denote the i'th block by X_i. We claim that there has to be an index $1 \leq j \leq t$ such that the blocks $(X_1 \circ \ldots \circ X_j)$ and $(X_{j+1} \circ \ldots \circ X_t)$ are a 2-block-source with min-entropy $k/4t \approx \delta^2 n$. To meet the definition of block-sources, we need to pad the two blocks above so that they will be of length n, but we ignore such technicalities in this overview.

To see why something like this must be true, let us consider the case of Shannon entropy. For Shannon entropy we have the chain rule $H(X) = \sum_{1 \leq j \leq t} H(X_j | X_1, \ldots, X_{j-1})$. Imagine going through the blocks one by one and checking whether the conditional entropy of the current block is at least $k/4t$. Since the total entropy is at least k, we must find such a block j. Furthermore, this block is of length n/t and so has entropy at most $n/t < k/10$. It follows that the total entropy we've seen this far is bounded by $t \cdot (k/4t) + k/10 < k/2$. This means that the remaining blocks must contain the remaining $k/2$ bits of entropy even when conditioned on the previous blocks.

Things become aren't so straightforward when dealing with min-entropy instead of entropy. Unlike Shannon entropy, min-entropy does not have a chain rule and the claim above does not hold in analogy. Nevertheless, imitating the argument above for min-entropy gives that for any k-source X there exists a small deficiency subsource X' of X such that there exist an index j for which the blocks $(X'_1 \circ \ldots \circ X'_j$ and $X'_{j+1} \circ \ldots \circ X'_t$ are a 2-block-source with min-entropy $k/4t \approx \delta^2 n$. As we explained earlier, this is helpful for constructing dispersers as we can forget about the initial source X and restrict our attention to the "nicely structured" subsource X'.

However, note that in order to use our extractors from Theorem 2.4 we need to also find the index j. This seems very hard as the disperser we are constructing is only given one sample x out of the source X and it seems impossible to use this information in order to find j. Moreover, the same sample x can appear with positive probability in many different sources that have different values of j.

2.4 Identifying high entropy parts in the source

Barak et al. [BKS+05] devised a technique which they call "the challenge-response mechanism" that in some sense allows the disperser to locate the high entropy block X_j in the source X. This method also relies on the other k-source Y. An important contribution of this paper is improving their method and extending it to detect blocks with much lower entropy. We will not attempt to describe how this method works within this informal overview as the technique is somewhat complicated and it is hard to describe it without delving into details. We do give a more detailed informal description (that still avoids many technical issues) in the informal explanation in Section 5.

In this high level overview we will only explain in what sense the challenge-response method finds the index j. Let us first recall the setup. The disperser obtains two inputs x and y from two independent k-sources X and Y. We have that X has a subsource X' which is a block-source. More precisely, there exists an index j such that the blocks $(X'_1 \circ \ldots \circ X'_j)$ and $(X'_{j+1} \circ \ldots \circ X'_t)$ are a 2-block-source with min-entropy $k/4t \approx \delta^2 n$.

Using the challenge-response mechanism, one can explicitly construct a function $FindIndex(x, y)$ such that there exist low deficiency subsources X^{good} of X' and Y^{good} of Y such that:

- $FindIndex(X^{\mathsf{good}}, Y^{\mathsf{good}})$ outputs the correct index j (with high probability).
- X^{good} is a 2-block-source according to the index j above.

Loosely speaking, this means that we can restrict our attention (in the analysis) to the independent sources $X^{\mathsf{good}}, Y^{\mathsf{good}}$. These sources are sources from which we can extract randomness! More precisely, when given x, y that are sampled from these sources we can compute $FindIndex(x, y)$ and then run the extractor from Theorem 2.4 on x, y using the index $FindIndex(x, y)$. The properties above guarantee that we have a positive probability to output both zero and one, which ensures that our algorithm is a disperser.

2.5 On extending this argument to $k < \sqrt{n}$

In the informal discussion above we only handled the case when $k = \delta n$ for some constant $\delta > 0$, though in this paper we are able to handle the case of $k = n^{o(1)}$. It turns out that there are several obstacles that need to be overcome before we can apply the strategy outlined above when $k < \sqrt{n}$.

Existence of block-sources in general sources The method we used for arguing that every k-source has a subsource which is a 2-block-source does not work when $k < \sqrt{n}$. If we partition the source X into $t < \sqrt{n}$ blocks then the length of each block is $n/t > k$ and it could be the case that all the entropy lies in one block (and in that case the next blocks contain no conditional entropy). On the other hand if we choose $t > \sqrt{n}$ then our analysis only gives a block-source with entropy $k/4t < 1$ which is useless.

In order to handle this problem we use a "win-win" case analysis (which is somewhat similar to the technique used in [RSW00]). We argue that either the source X has a subsource X' such that partitioning X' according to an index j gives a 2-block-source with min entropy $\approx k/c$, or there must exist a block j which has entropy larger than $\approx k/c$. We now explain how to handle the second case, ignoring for now the issue of distinguishing which of the cases we are in. Note that we partition X into t blocks of length n/t and therefore in the second case there is a block j where the min-entropy rate (i.e. the ratio of the min-entropy to the length) increased by a multiplicative factor of $t/c \gg 1$.

Loosely speaking, we already have a way to locate the block j (by using the challenge-response mechanism) and once we find it we can recursively call the disperser construction on x_j and y. Note that we are indeed making progress as the min-entropy rate is improving and it can never get larger than one. This means that eventually the min-entropy rate will be so high that we are guaranteed to have a block source, that we know how to handle.

It is also important to note that this presentation is oversimplified. In order to perform the strategy outlined above we need to also be able to distinguish between the case that our source contains a block-source and the case where it has a high entropy block. For this purpose we develop a new and more sensitive version of the challenge-response mechanism that is also able to distinguish between the two cases.

Lack of somewhere extractors for low entropy In this high level overview we did not discuss the details of how to implement the function $FindIndex$ using the challenge-response mechanism. Still, we remark that the implementation in [BKS$^+$05] relies on certain objects called "somewhere extractors". While we do not define these objects here (the definition can be found in the formal sections), we mention that we do not know how to construct these objects for $k < \sqrt{n}$. To address this problem we implement the challenge-response mechanism in a different way relying only on objects that are available in the low-entropy case.

3 Preliminaries

The following are some definitions and lemmas that are used throughout this paper.

3.1 Basic notations and definitions.

Often in technical parts of this paper, we will use constants like 0.9 or 0.1 where we could really use any sufficiently large or small constant that is close to 1 or 0. We do this because it simplifies the presentation by reducing the number of additional variables we will need to introduce.

In informal discussions throughout this paper, we often use the word *entropy* losely. All of our arguments actually involve the notion of *min-entropy* as opposed to Shannon entropy.

Random variables, sources and min-entropy. We will usually deal with random variables which take values over $\{0,1\}^n$. We call such a random variable an *n-bit source*. The *min-entropy* of a random variable X, denoted by $H_\infty(X)$, is defined to equal $\min_x\{-\log_2(\Pr[X=x])\}$, or equivalently $\log_2(1/\max_x\{\Pr[X=x]\})$. If X is an n-bit source with $H_\infty(X) \geq k$ and n is understood from the context then we'll call X a *k-source*.

Definition 3.1 (Statistical distance.)**.** If X and Y are random variables over some universe U, the *statistical distance* of X and Y, denoted by $|X-Y|$ is defined to be $\frac{1}{2}\sum_{u\in U}|\Pr[X=u]-\Pr[Y=u]|$.

We have the following simple lemma:

Lemma 3.2 (Preservation of strongness under convex combination)**.** *Let X, O, U, Q be random variables over the same finite probability space, with U, O both random variables over $\{0,1\}^m$. Let $\epsilon_1, \epsilon_2 < 1$ be constants s.t. :*

$$\Pr_{q\leftarrow_R Q}[|(X|Q=q)\circ(O|Q=q)-(X|Q=q)\circ(U|Q=q)| \geq \epsilon_1] < \epsilon_2$$

i.e. conditioned on Q being fixed and good*, $X\circ O$ is statistically close to $X\circ U$.*

Then we get that $|X\circ O - X\circ U| < \epsilon_1+\epsilon_2$.

Definition 3.3 (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 exists a set $A \subseteq \{0,1\}^n$ such that $(X|A) = X'$ and $\Pr[X\in A] \geq 2^{-d}$.

Definition 3.4 (Block-sources)**.** A distribution $X = X^1\circ X^2\circ\cdots\circ X^{\mathsf{C}}$ is called a $(k_1,k_2,\ldots,k_{\mathsf{C}})$-block-source if for all $i = 1,\ldots,\mathsf{C}$, we have that for all $x_1\in X^1,\ldots,x_{i-1}\in X^{i-1}$, $H_\infty(X^i|X^1 = x_1,\ldots,X^{i-1}=x_{i-1}) \geq k_i$, i.e., each block has high min-entropy even conditioned on the previous blocks. If $k_1 = k_2 = \cdots = k_{\mathsf{C}} = k$, we say that X is a k-block-source.

Definition 3.5 (Somewhere Random Sources)**.** A source $X = (X_1,...,X_t)$ is $(t\times r)$ *somewhere-random* if each X_i takes values in $\{0,1\}^r$ and there is an i such that X_i is uniformly distributed.

Definition 3.6. We will say that a collection of somewhere-random sources is *aligned* if there is some i for which the i'th row of *every* SR-source in the collection is uniformly distributed.

Since we shall have to simultaneously use the concept of block-sources and somewhere random sources, for clarity we use the convention that the word *block* refers to a part of a block-source. The word *row* will be used to refer to a part in a somewhere random source.

Definition 3.7 (Weak somewhere random sources)**.** A source $X = (X_1,\ldots,X_t)$ 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.

Often we will need to apply a function to each row of a somewhere source. We will adopt the following convention: if $f : \{0,1\}^r \times \{0,1\}^r \to \{0,1\}^m$ is a function and a, b are samples from $(t \times r)$ somewhere sources, $f(\vec{a}, \vec{b})$ refers to the $(t \times m)$ string whose first row is obtained by applying f to the first rows of a, b and so on. Similarly, if a is an element of $\{0,1\}^r$ and b is a sample from a $(t \times r)$ somewhere source, $f(a, \vec{b})$ refers to the $(t \times m)$ matrix whose ith row is $f(a, b_i)$.

Many times we will treat a sample of a somewhere random source as a set of strings, one string from each row of the source.

Definition 3.8. Given ℓ strings of length n, $x = x_1, \ldots, x_\ell$, define $\mathsf{Slice}(x, w)$ to be the string $x' = x'_1, \ldots, x'_\ell$ such that for each i x'_i is the prefix of x_i of length w.

3.1.1 Extractors, dispersers and their friends.

In this section we define some of the objects we will later use and construct. All of these objects will take two inputs and produce one output, such that under particular guarantees on the distribution of the input, we'll get some other guarantee on the distribution of the output. Various interpretation of this vague sentence lead to extractors, dispersers, somewhere extractors, block extractors etc..

Definition 3.9 (Two-source extractor)**.** Let $n_1, n_2, k_1, k_2, m, \epsilon$ be some numbers. A function $\mathsf{Ext} : \{0,1\}^{n_1 \times n_2} \to \{0,1\}^m$ is called an *2-source extractor with k_1, k_2 min-entropy requirement, n-bit input, m-bit output and ϵ-statistical distance* if for every independent sources X and Y over $\{0,1\}^{n_1}$ and $\{0,1\}^{n_2}$ respectively satisfying

$$H_\infty(X) \geq k_1 \quad \text{and} \quad H_\infty(Y) \geq k_2 \tag{1}$$

it holds that

$$\left| \mathsf{Ext}(X, Y) - U_m \right| \leq \epsilon \tag{2}$$

In the common case of a *seeded extractor* we have $n_2 = k_2$ (and hence the second input distribution is required to be uniform). A non-trivial construction will satisfy of course $n_2 \ll m$ (and hence also $n_2 \ll k_1 < n$). Thus, two source extractors are strictly more powerful than seeded extractor. However, the reason seeded extractors are more popular is that they suffice for many applications, and that (even after this work) the explicit construction for seeded extractors have much better parameters than the explicit constructions for 2-source extractors with $k_1 \ll n_1$, $k_2 \ll n_2$. (Note that this is not the case for *non-explicit* construction, where 2-source extractors with similar parameters to the best possible seeded extractors can be shown to exist using the probabilistic method.)

Variants. We'll use many variants of extractors in this paper to various similar combinatorial objects. Most of the variants are obtained by giving different the conditions on the input (Equation 1) and the guarantee on the output (Equation 2). Some of the variants we will consider will be:

Dispersers. In dispersers, the output guarantee (Equation 2) is replaced with $|\mathsf{Supp}(\mathsf{Ext}(X, Y))| \geq (1 - \epsilon)2^m$,

Somewhere extractors. In *somewhere extractors* the output guarantee (Equation 2) is replaced with the requirement that $|\mathsf{Ext}(X, Y) - Z| < \epsilon$ where Z is a *somewhere random source* of $t \times m$ rows for some parameter t.

Extractors for block-sources. In *extractors for block-sources* the input requirement (Equation 1) is replaced with requirement that X and Y are *block-sources* of specific parameters. Similarly we will define extractors for other families of inputs (i.e. somewhere random sources) and extractors where each input should come from a different family.

Strong extractors. Many of these definition have also a *strong* variant, and typically constructions for extractors also achieve this strong variant. An extractor is *strong in the first input* if the output requirement (Equation 2) is replaced with $|(X, \mathsf{Ext}(X, Y)) - (X, U_m)| \leq \epsilon$. Intuitively this condition means that the output is uniform even on conditioning X. We define an extractor to be strong in the second input similarly. If the extractor is strong in both inputs, we simply say that it is *strong*.

Remark 3.10 (Input lengths)**.** Whenever we have a two source extractor $\mathsf{Ext} : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \to \{0,1\}^m$ with inputs lengths n_1, n_2 and entropy requirement k_1, k_2 we can always invoke it on shorter sources with the same entropy, by simply padding it with zeros. In particular if we have an extractor with $n_1 = n_2$ we can still invoke it on inputs of *unequal* length by padding one of the inputs. The same observation holds for the other source types we'll use, namely *block* and *somewhere random* sources, if the padding is done in the appropriate way (i.e., pad each block for block-sources, add all zero rows for somewhere random sources), and also holds for all the other extractor-like objects we consider (dispersers, somewhere extractors, and their subsource variant). In the following, whenever we invoke an extractor on inputs shorter than its "official" input length, this means that we use such a padding scheme.

3.2 Useful facts and lemmas.

Fact 3.11. *If X is an (n, k)-source and X' is a deficiency d subsource of X then X' is an $(n, k-d)$-source.*

Fact 3.12. *Let X be a random variable with $H_\infty(X) = k$. Let A be any event in the same probability space. Then*

$$H_\infty(X|A) < k' \Rightarrow \Pr[A] < 2^{k'-k}$$

3.2.1 Fixing functions and projections.

Given a source X over $\{0,1\}^n$ and a function $F : \{0,1\}^n \to \{0,1\}^m$, we often will want to consider subsources of X where F is fixed to some value, and provide some bounds on the deficiency. Thus, the following lemma would be useful:

Lemma 3.13 (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. Then there exists $a \in \{0,1\}^m$ and a deficiency m subsource X' of X such that $F(x) = a$ for every x in X'. Furthermore, for every $a \in \mathsf{Supp}(F(X))$ let X_a be the subsource of X defined by conditioning on $F(X) = a$. Then, if we choose a at random from the source $F(X)$ then with probability $\geq 1 - 2^{-\ell}$, the deficiency of X_a is at most $m + \ell$.*

Proof. Let $\ell > 0$ be some number and let A be the set of $a \in \{0,1\}^m$ such that $\Pr[F(x) = a] < 2^{-m-\ell}$. Since $|A| \leq 2^m$, we have that $\Pr[F(X) \in A] < 2^{-\ell}$. If we choose $a \leftarrow_{\mathrm{R}} F(X)$ and $a \notin A$, we get that $X|F(X) = a$ has deficiency $\leq m + \ell$. Choosing $\ell = 0$ we get the first part of the lemma, and choosing $\ell = m$ we get the second part. □

The following lemma will also be useful:

Lemma 3.14 (Fixing a few bits in X)**.** *Let X be an (n,k) source. Let $S \subseteq [n]$ with $|S| = n - n'$. Let $X_{|S}$ denote the projection of X to the bit locations in S. Then for every l, $X_{|S}$ is 2^{-l}-close to a $(n-n', k-n'-l)$ source.*

Proof. Let $\overline{S}$ be the complement of S.

Then $X_{|S}$ is a convex combination over $X_{|\overline{S}}$. For each setting of $X_{|\overline{S}} = h$, we condition the distribution $X_{|S}|(X_{|\overline{S}} = h)$.

Define $H = \{h \in \{0,1\}^{n'} | H_\infty(X_{|S}|X_{|\overline{S}} = h) < n' + k - l\}$. Notice that $H_\infty(X_{|S}|X_{|\overline{S}} = h) = H_\infty(X|X_{|\overline{S}} = h)$. Then by Fact 3.12, for every $h \in H$, $\Pr[X_{|\overline{S}} = h] < 2^{k-n'-l-k} = 2^{-(n'+l)}$. Since $|H| \le 2^{n'}$, by the union bound we get that $\Pr[X_{|\overline{S}} \in H] \le 2^{-l}]$. □

In some situations we will have a source that is statistically close to having high min-entropy, but not close enough. We can use the following lemma to lose something in the entropy and get 0 error on some subsource.

Lemma 3.15. *Let X be a random variable over $\{0,1\}^n$ s.t. X is ϵ-close to an (n,k) source, with $\epsilon \le 1/4$. Then there is a deficiency 2 subsource $X' \subseteq X$ s.t. X' is a $(n, k-3)$ source.*

Proof. Let t be a parameter that we will pick later. Let $H \subseteq \mathsf{Supp}(X)$ be defined as $H = \{x \in \mathsf{Supp}(X) | \Pr[X = x] > 2^{-t}\}$. H is the set of heavy points of the distribution X. By the definition of H, $|H| \le 2^t$.

Now we have that $\Pr[X \in H] - 2^{-k}|H| \le \epsilon$, since X is ϵ-close to a source with min-entropy k. This implies that $\Pr[X \in H] \le \epsilon + 2^{-k}|H| \le \epsilon + 2^{t-k}$.

Now consider the subsource $X' \subseteq X$ defined to be $X|X \in (\mathsf{Supp}(X) \setminus H)$. For every $x \in \mathsf{Supp}(X')$, we get that

$\Pr[X' = x] = \Pr[X = x|X \notin H] \le \frac{\Pr[X=x]}{\Pr[X \notin H]} \le \frac{2^{-t}}{1-(\epsilon+2^{t-k})}$

Setting $t = k-2$, we get that $\Pr[X' = x] \le \frac{2^{-k+2}}{1-(\epsilon+2^{-2})} \le 2^{-k+3}$. □

3.2.2 Convex combinations.

Definition 3.16 (Convex combination)**.** Let X be a random variable and let $\{Y_i\}_{i \in U}$ be a family of random variables indexed by an element in some universe U. We say that X is a convex combination of the family $\{Y_i\}$ if there exists a random variable I over U such that $X = Y_I$.

A key observation that is essential to our results is that random variables that are convex combinations of sources with some good property are usually good themselves. This is captured in the following easy propositions:

Proposition 3.17. *Let X, Z be random variables s.t. X is a convex combination of sources which are ϵ-close to Z. Then X is ϵ-close to Z.*

3.2.3 Conditional entropy.

If $X = X_1 \circ \cdots \circ X_{\mathsf{t}}$ is a random variable (not necessarily a block-source) over $\{0,1\}^n$ divided into t blocks in some way, and $x_1, \ldots, x_i$ are some strings with $0 \le i < \mathsf{t}$, we use the notation $X|x_1, \ldots, x_i$ to denote the random variable X conditioned on $X_1 = x_1, \ldots, X_i = x_i$. For $1 \le i < j \le \mathsf{t}$, we denote by $X_{i,\ldots,j}$ the projection of X into the blocks $X_i, \ldots, X_j$. We have the following facts about such sources:

Lemma 3.18 (Typical prefixes). *Let $X = X_1 \circ \cdots \circ X_{\mathsf{t}}$ be a random variable divided into t blocks, let $X' = X|A$ be a deficiency d subsource of X, and let ℓ be some number. Then for every $1 \le i \le \mathsf{t}$, with probability at least $1 - 2^{-\ell}$, a random prefix $x_1, \ldots, x_i$ in X' satisfies $\Pr[X \in A|x_1, \ldots, x_i] \ge 2^{-d-\ell}$.*

Proof. We denote by X^1 the first i blocks of X. Let B be the event determined by X^1 that $\Pr[X \in A|X^1] < 2^{-d-\ell}$. We need to prove that $\Pr[B|A] < 2^{-\ell}$ but this follows since $\Pr[B|A] = \frac{\Pr[A \cap B]}{\Pr[A]} \le 2^d \Pr[A \cap B]$. However $\Pr[A \cap B] \le \Pr[A|B] = \sum_{x \in B} \Pr[A|X^1 = x] Pr[X^1 = x|B] < 2^{-d-\ell}$. ☐

As a corollary we get the following

Corollary 3.19 (Subsource of block-sources). *Let $X = X_1 \circ \cdots \circ X_{\mathsf{C}}$ be a k-entropy C-block-source (i.e., for every $x_1, \ldots, x_i \in \mathsf{Supp}(X_{1,\ldots,i})$, $H_\infty(X_{i+1}|X_{1,\ldots,i} = x_1, \ldots, x_i) > k$) and X' be a deficiency d subsource of X. Then X' is $\mathsf{C}2^{-l}$ statistically close to being a $k - d - l$ C-block-source.*

Proof. Let $X' = X|A$ and define B to be following the event over X': $x = x_1, \ldots, x_{\mathsf{C}} \in B$ if for some $i \in [\mathsf{C}]$, $\Pr[X \in A|x_1, \ldots, x_i] < 2^{-d-l}$. By Lemma 3.18, $\Pr[X' \in B] < \mathsf{C}2^{-l}$. However, for every $x = x_1, \ldots, x_{\mathsf{C}} \in \bar{B} = A \setminus B$, we get that $Y' = X'_{i+1}|x_1, \ldots, x_{i-1}$ is a source with

$H_\infty(Y') \ge H_\infty(Y) - d - l \ge k - d - l$. Hence $X'|\bar{B}$ is a $k - d - l$-block-source of distance $\mathsf{C}2^{-l}$ from X'. ☐

If $X = X_1 \circ \cdots \circ X_{\mathsf{t}}$ is a source divided into t blocks then in general, the entropy of X_i conditioned on some prefix $x_1, \ldots, x_{i-1}$ can depend on the choice of prefix. However, the following lemma tells us that we can restrict to a low deficiency subsource on which this entropy is always roughly the same, regardless of the prefix. Thus we can talk about the conditional entropy of a block X_i without referring to a particular prefix of it.

Lemma 3.20 (Fixing entropies). *Let $X = X_1 \circ X_2 \circ \cdots \circ X_{\mathsf{t}}$ be a t-block random variable over $\{0,1\}^n$, and let $0 = \tau_1 < \tau_2 < \cdots < \tau_{c+1} = n$ be some numbers. Then, there is a deficiency $\mathsf{t}^2 \log c$ subsource X' of X and a sequence $\bar{e} = e_1, \ldots, e_{\mathsf{t}} \in [c]^{\mathsf{t}}$ such that for every $0 < i \le \mathsf{t}$ and every sequence $x_1, \ldots, x_{i-1} \in \mathsf{Supp}(X'_{1,\ldots,i-1})$, we have that*

$$\tau_{e_i} \le H_\infty(X'_i|x_1, \ldots, x_{i-1}) \le \tau_{e_i+1} \tag{3}$$

Proof. We prove this by induction. Suppose this is true for up to $\mathsf{t} - 1$ block 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 $X_{2,\ldots,i}|x_1$. By the induction hypothesis there exists a $(\mathsf{t}-1)^2 \log c$ deficiency subsource $Y'(x_1)$ of $Y(x_1)$ source and $\bar{e}(x_1) \in [c]^{\mathsf{t}-1}$ the sequence such that $Y'(x_1)$ satisfies Equation 3 with respect to $\bar{e}(x_1)$. Define the function $f : X_1 \to [c]^{\mathsf{t}-1}$ that maps x_1 to $\bar{e}(x_1)$ and pick a subsource X'_1 of X_1 of deficiency $(\mathsf{t}-1) \log c$ such that f is constant on X'_1. That is, there are some values $e_2, \ldots, e_{\mathsf{t}} \in [c]^{\mathsf{t}-1}$ such that $F(x_1) = e_2, \ldots, e_{\mathsf{t}}$ with probability 1. We let the source X' be X conditioned on the event that $X_1 \in \mathsf{supp}(X'_1)$ and $X_2, \ldots, X_{\mathsf{t}} \in \mathsf{supp}(Y(X_1))$.

The deficiency of X' is indeed at most $(\mathsf{t} - 1) \log c + (\mathsf{t} - 1)^2 \log c < \mathsf{t}^2 \log c$. ☐

Corollary 3.21. *If X in the lemma above is a k-source, and $\bar{e}$ is as in the conclusion of the lemma, we must have that $\sum_{i=1}^{t} \tau_{e_i+1} \ge k - t^2 \log c$.*

Proof. If this was not the case, we could find some string in the support of X which is too heavy (simply take the heaviest string allowed in each successive block). ☐

Proposition 3.22. *Let $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ be a seeded (n, k, ϵ) strong extractor. Let X be any (n, k) source. Let $\{0,1\}^d = \{s_1, s_2, \ldots, s_{2^d}\}$. Then $\mathsf{Ext}(X, s_1) \circ \mathsf{Ext}(X, s_2) \circ \cdots \circ \mathsf{Ext}(X, s_{2^d})$ is ϵ-close to a $(2^d \times m)$ somewhere-random source.*

Proof. This follows immediately from the definition of a strong seeded extractor (Definition 3.9). ☐

3.3 Some results from previous works.

We'll use the following results from some previous works.

Theorem 3.23. *[LRVW03] For any constant $\alpha \in (0,1)$, every $n \in \mathbb{N}$ and $k \leq n$ and every $\epsilon \in (0,1)$ where $\epsilon > exp(-\sqrt{k})$, there is an explicit (k, ϵ) seeded extractor* $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^{O(\log n + \log(n/k)\log(1/\epsilon))} \to \{0,1\}^{(1-\alpha)k}$.

Theorem 3.24 ([Tre01, RRV02])**.** *For every $n, k \in \mathbb{N}$, $\epsilon > 0$, there is an explicit (n, k, ϵ)-strong extractor* $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^{k - O(\log^3(n/\epsilon))}$ *with $d = O(\log^3(n/\epsilon))$.*

Theorem 3.25 ([CG88, Vaz85])**.** *For all $n, \delta > 0$, there exists a polynomial time computable strong extractor* $\mathsf{Vaz} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$ *with $m = \Omega(n)$ and error $\epsilon = 2^{-\Omega(n)}$.*

Theorem 3.26 ([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 3.27 ([Rao06])**.** *There is a polynomial time computable strong extractor* $\mathsf{2SRExt} : \{0,1\}^{tn} \times \{0,1\}^{tn} \to \{0,1\}^m$ *such that for every constant $\gamma < 1$ and n, t with $t = t(n)$, $t < n^\gamma$ there exists a constant $\alpha(\gamma) < 1$ such that* $\mathsf{2SRExt}$ *succeeds as long as X is a $(t \times n)$ SR-source and Y is an independent aligned $(t \times n)$ SR-source, with $m = n - n^\alpha$ and error $2^{-n^{1-\alpha}}$.*

4 Ingredients

In this section we describe two new ingredients that are used in our construction.

4.1 Extractor for one block-source and one general source

We construct an extractor that works for two sources, given an assumption on one of the sources. The assumption is that the first source is a *block-source*, which means that it is divided into C blocks such that each block has entropy above a certain threshold *even conditioned on all previous blocks.* As mentioned in the introduction, block-sources turn out to be very useful in many settings in randomness extraction. Rao [Rao06] gave an extractor for 2 independent block-sources with few blocks. Here we improve his construction in two ways, both of which are important for the application to our disperser construction.

- We relax the hypothesis so that we need only one block-source. The second source can be arbitrary.
- We improve the error of the construction from $1/\mathrm{poly}(k)$ to $2^{-k^{\Omega(1)}}$.

We will prove the following theorem (which is a formal restatement of Theorem 2.4).

Theorem 4.1 (Block + Arbitrary Source Extractor). *There are constants c_1, c_2 and a polynomial time computable function* $\mathsf{BExt} : \{0,1\}^{\mathsf{C}n} \times \{0,1\}^n \to \{0,1\}^m$ *such that for every n, k, with $k > \log^{10}(n)$ with $\mathsf{C} = O(\frac{\log n}{\log k})$, if $X = X^1 \circ \cdots \circ X^{\mathsf{C}}$ is a k-block-source and Y is an independent k-source*

$$|\mathsf{BExt}(X, Y) - U_m| < 2^{-k^{c_1}}$$

with $m = c_2 k$.

The low error guaranteed by this theorem is important for applications that require a *negligible* error. Since the concatenation of independent sources is a block-source, an immediate corollary of the above theorem is a new extractor for independent sources with exponentially small error (the corollary below is a formal restatement of Theorem 1.6).

Corollary 4.2 (Independent Source Extractor). *There are constants c_1, c_2 and a polynomial time computable function* $\mathsf{BExt} : (\{0,1\}^n)^{\mathsf{C}} \to \{0,1\}^m$ *such that for every $n, , k$, with $k > \log^{10}(n)$ with $\mathsf{C} = O(\frac{\log n}{\log k})$, if $X^1, \ldots, X^{\mathsf{C}}$ are independent (n, k) sources,*

$$|\mathsf{BExt}(X_1, \ldots, X_{\mathsf{C}}) - U_m| < 2^{-k^{c_1}}$$

with $m = c_2 k$.

The proof of Theorem 4.1 appears in Section 7.

4.2 A 2-Source Somewhere Extractor with exponentially small error

A technical tool that we will need is a somewhere extractor from 2 independent sources which has a polynomial number of output rows, but exponentially small error. This will be used to generate the *responses* throughout our disperser construction. Note that we can get a polynomial number of output rows by using a seeded extractor with just one of the sources, but in this case the errror would not be small enough. In addition, in this section we will prove some other technical properties of this construction which will be critical to our construction.

Theorem 4.3 (Low Error Somewhere Extractor). *There is a constant γ and a polynomial time computable function* $\mathsf{SE} : (\{0,1\}^n)^2 \to (\{0,1\}^m)^l$ *such that for every $n, k(n) > \log^{10} n, \log^4 n < m < \gamma k$ and any two (n, k) sources X, Y, we have:*

- ***Few rows*** $l = \mathrm{poly}(n)$.
- ***Small error*** $\mathsf{SE}(X, Y)$ *is 2^{-10m}-close to a convex combination of somewhere random distributions and this property is strong with respect to both X and Y. Formally:*

$$\Pr_{y \leftarrow_R Y}[\mathsf{SE}(X, y) \text{ is } 2^{-10m}\text{-close to being } SR\ ] > 1 - 2^{-10m}$$

- ***Hitting strings*** *Let c be any fixed m bit string. Then there are subsources $\hat{X} \subset X, \hat{Y} \subset Y$ of deficiency $2m$ and an index i such that* $\Pr[c = \mathsf{SE}(\hat{X}, \hat{Y})_i] = 1$.
- ***Fixed rows on low deficiency subsources*** *Given any particular row index i, there is a subsource $(\hat{X}, \hat{Y}) \subset (X, Y)$ of deficiency $20m$ such that $\mathsf{SE}(\hat{X}, \hat{Y})_i$ is a fixed string. Further, (X, Y) is 2^{-10m}-close to a convex combination of subsources such that for every $(\hat{X}, \hat{Y})$ in the combination,*

– $\hat{X}, \hat{Y}$ *are independent.*

– $\mathsf{SE}(\hat{X}, \hat{Y})_i$ *is constant.*

The **{Hitting strings}** and **{Fixed rows on low deficiency subsources}** properties may at first seem quite similar. The difference is in the quantifiers. The first property guarantees that for *every string* c, we can move to low deficiency subsources such there *exists an index* in the output of SE where the string is seen with probability one. The second property guarantees that for *every index* i, we can move to low deficiency subsources where the output in that index is fixed to *some string.*

Proof. The algorithm SE is the following:

Algorithm 4.4.
$\mathsf{SE}(x, y)$
Input: x, y samples from two independent sources with min-entropy k.
Output: A $\ell \times m$ boolean matrix.
Subroutines:

- A seeded extractor Ext with $O(\log n)$ seed length (for example by Theorem 3.23), setup to extract from entropy threshhold $0.9k$, with output length m and error $1/100$.
- The extractor Raz from Theorem 3.26, setup to extract m bits from an (n, k) source using a weak seed of length m bits with entropy $0.9m$. We can get such an extractor with error 2^{-10m}.

1. For every seed i to the seeded extractor Ext, output $\mathsf{Raz}(x, \mathsf{Ext}(y, i))$.

We will prove each of the items in turn.

- **Few rows** By construction.
- **Small error** We will argue that the strong error with respect to Y is small. Consider the set of bad y's,

$$B = \{y : \forall i |\mathsf{Raz}(X, \mathsf{Ext}(y, i)) - U_m| \geq 2^{-\gamma' k}\}$$

where here γ' is the constant that comes from the error or Raz's extractor.

We would like to show that this set is very small.

Claim 4.5. $|B| < 2^{0.9k}$

Suppose not. Let B denote the source obtained by picking an element of this set uniformly randomly. Since Ext has an entropy threshhold of $0.9k$, there exists some i such that $\mathsf{Ext}(B, i)$ is $1/100$ close to uniform. In particular, $|\mathsf{Supp}(\mathsf{Ext}(B, i))| \geq 0.992^m > 2^{0.9m}$. This is a contradiction, since at most $2^{0.9m}$ seeds can be bad for Raz.

Thus we get that

$$\Pr_{y \leftarrow_{\mathrm{R}} Y}[|\mathsf{Raz}(X, \mathsf{Ext}(y, i)) - U_m| < 2^{-k^{\gamma'}}] < 2^{0.9k}/2^k = 2^{-0.1k}$$

Setting $\gamma = \gamma'/10$, we get that $10m < 10\gamma k < \gamma' k$ and $10m < 0.1k$.

- **Hitting strings** The proof for this fact follows from the small error property. Let $\tilde{Y} = Y|(Y \notin B)$, where B is the set of bad y's from the previous item. Then we see that for every $y \in \mathsf{Supp}(\tilde{Y})$, there exists some i such that $|\mathsf{Raz}(X, \mathsf{Ext}(y,i)) - U_m| < 2^{-10m}$. By the pigeonhole principle, there must be some seed s and some index i such that:

$$\Pr_{y \leftarrow_{\mathrm{R}} \tilde{Y}}[\mathsf{Ext}(y,i) = s] \geq \frac{1}{l2^m}$$

  Fix such an i and string s and let $\hat{Y} = \tilde{Y}|\mathsf{Ext}(\tilde{Y}, i) = s$. This subsource has deficiency at most $1 + m + \log l < 2m$ from Y. Thus $\mathsf{Ext}(\hat{Y}, i)$ is fixed and $|\mathsf{Raz}(X, \mathsf{Ext}(y,i)) - U_m| < 2^{-10m}$. Note that the i'th element of the output of $\mathsf{SE}(X, \hat{Y})$ is a function only of X. Thus we can find a subsource $\hat{X} \subset X$ of deficiency at most $2m$ and string $c \in \{0,1\}^m$ such that $\mathsf{SE}(\hat{X}, \hat{Y})_i = c$.

- **Fixed rows on low deficiency subsources** Let i be any fixed row. For any m-bit string a, let $Y_a \subset Y$ be defined as $Y|(\mathsf{Ext}(Y,i) = a)$. By Lemma 3.13, for any $\ell > 1$, $\Pr_{a \leftarrow_{\mathrm{R}} \mathsf{Ext}(Y,i)}[Y_a$ has deficiency more than $m + \ell\ ] < 2^{-\ell}$.

  Let $A = \{a : Y_a$ has deficiency more than $m + \ell\}$. Then, by Lemma 3.13, we see that Y is $2^{-\ell}$-close to a source $\overline{Y}$, where $\Pr[\mathsf{Ext}(\overline{Y}, i) \notin A] = 1$, and $\overline{Y}$ has min-entropy at least $k - 1$.

  We break up $\overline{Y}$ into a convex combination of variables $\hat{Y}_a = \overline{Y}|(\mathsf{Ext}(\overline{Y}, i) = a)$, each of deficiency at most $m + \ell$.

  Similarly we can argue that X is $2^{-\ell}$-close to a random variable $\overline{X}_a$ with min-entropy $k - 1$, where $\overline{X}_a$ is a convex combination of subsources $\hat{X}_{a,b}$ with deficiency at most $m + \ell$ such that $\mathsf{Raz}(\hat{X}_{a,b}, a)$ is constant and equal to b.

  Thus we obtain our final convex combination. Each element $\hat{X}_{a,b}, \hat{Y}_b$ of the combination is associated with a pair (a, b) of m bit strings. By construction we see that the i'th row $\mathsf{SE}(\hat{X}_{a,b}, \hat{Y}_b)_i = a$ and that $\hat{X}_{a,b}, \hat{Y}_b$ each have min-entropy $k - m - \ell$.

■

5 Informal overview of the construction and analysis of the disperser

In this section we give a detailed yet informal description of our construction and analysis. On the one hand this description presents the construction and steps of the analysis in a very detailed way. On the other hand, the fact that this section is not a formal proof allows us to abstract many tedious technical details and parameters and to focus only on what we consider to be central. This section complements Section 2 and provides a detailed explanation of the challenge-response mechanism.

The structure of this presentation closely imitates the way we formally present our construction and proof in Section 6 and we make use of "informal lemmas" in order to imitate the formal presentation and make the explanation more clear. We stress that these "informal lemmas" should not be interpreted as formally claimed by this paper (and these lemmas typically avoid being precise regarding parameters). We furthermore stress that the content of this section is not used in the latter part of the paper and the reader may safely skip to the formal presentation in Section 6 if she wishes.

The setup We are given two input sources X, Y which have some min-entropy and would like to output a non-constant bit.

The idea behind the construction is to try to convert the first source X into a block-source or at least find a subsource (Definition 3.3) $X^{\mathsf{good}} \subset X$ which is a block-source. Once we have such a block-source, we can make use of some of the technology we have developed for dealing with block-sources (for instance the extractor BExt of Theorem 4.1).

One problem with this approach is that there is no deterministic procedure that transforms a source into a block-source, or even to a short (e.g. of length much less than $\frac{n}{k}$) list of sources, one of which is guaranteed to be a block-source. Still, as we will explain shortly, we will manage to use the second source Y to "convert" X into a block-source. Loosely speaking, we will show that there exist independent subsources $X^{\mathsf{good}} \subset X$ and $Y^{\mathsf{good}} \subset Y$ such that X^{good} is a block-source and our construction "finds" this block-source when applied on $X^{\mathsf{good}}, Y^{\mathsf{good}}$. This task of using one source to find the entropy in the other source while maintaining independence (on subsources) is achieved via the "challenge-response mechanism".

We describe our construction in two phases. As a warmup, we first discuss how to use the challenge-response mechanism in the case when the two sources have linear min-entropy (this was first done by Barak et al. [BKS$^+$05]). Then we describe how to adapt the challenge-response mechanism for the application in this paper.

5.1 Challenge-Response Mechanism for Linear Min-Entropy

The challenge-response mechanism was introduced in [BKS$^+$05] as a way to use one source of randomness to *find* the entropy in another source. Since they only constructed 2 source dispersers that could handle linear min-entropy, they avoided several complications that we will need to deal with here. Still, as an introduction to the challenge-response mechanism, it will be enlightening to revisit how to use the mechanism to get dispersers for linear min-entropy. Below we will give a sketch of how we might get such a disperser using the technology that is available to us at this point. Note that the construction we discuss here is slightly different from the one originally used by Barak et al.

We remind the reader again of the high level scheme of our construction. We will construct a polynomial time computable function Disp with the property that for any independent linear entropy sources X, Y, there exist subsources $X^{\mathsf{good}} \subset X, Y^{\mathsf{good}} \subset Y$ with the property that $\mathsf{Disp}(X^{\mathsf{good}}, Y^{\mathsf{good}})$ is both 0 and 1 with positive probability. Since $X^{\mathsf{good}}, Y^{\mathsf{good}}$ are subsources of the original sources, this implies that Disp is a disperser even for the original sources. Now let us describe the construction.

Let us assume that for linear min-entropy our extractor BExt requires only 2 blocks; so we have at our disposal a function $\mathsf{BExt} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ with the property that if $X_1 \circ X_2$ is a block-source with linear min-entropy, and Y is an independent block-source, $\mathsf{BExt}(X_1 \circ X_2, Y)$ is exponentially close to being a uniform bit.

We are given two sources X, Y which are independent sources with min-entropy δn, where δ is some small constant. We would be in great shape if we were given some additional advice in the form of an index $j \in [n]$ such that $X_{[j]} \circ X$ is a block-source with min-entropy say $\delta n/10$ (i.e. the first j bits of X have min-entropy $\delta n/10$ and conditioned on any fixing of these bits the rest of the source still has min-entropy at least $\delta n/10$). In this case we would simply use our block-source extractor BExt and be done. Of course we don't have any such advice, on the other hand, the good news is that it can be shown that such an index j *does* exist.

Step 1: Existence of a structured subsource We associate a *tree of parts* with the source X. In this warmup this will be a tree of depth 1, with the sample from X at the root of the tree. We break the sample from the source X into a constant $\mathsf{t} \gg 1/\delta$ number of equally sized parts $x = x_1, \ldots, x_{\mathsf{t}}$, each containing n/t consecutive bits. These are the children of the root. Our construction will now operate on the bits of the source that are associated with the nodes of this tree.

In the first step of the analysis, we use standard facts (Lemma 3.20 and Corollary 3.21) to show that:

Informal Lemma 5.1. *If X has min-entropy δn, there is a $j \in [t]$ and a subsource $\hat{X} \subset X$ in which:*

- *$\hat{X}_i$ is fixed for $i < j$.*
- *$H_\infty(\hat{X}_j) \geq \delta^2 n/10)$.*
- *$(\hat{X}_{j+1}, \ldots, \hat{X}_t)$ has conditional min-entropy at least $\delta^2 n/10$ given any fixing of $\hat{X}_j$.*

Given this lemma, our goal is to find this index j (which is the 'advice' that we would like to obtain). We will be able to do so on independent subsources of $\hat{X}$, Y. This is achieved via the challenge-response mechanism.

Step 2: Finding the structure using the challenge-response mechanism Here are the basic pieces we will use to find the index j:

1. A polynomial time computable function $\mathsf{Challenge} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^{\mathsf{clen}}$. In view of the final construction, we view the output of this function as a matrix with 1 row of length clen. We also require the following properties:

   Output length is much smaller than entropy $\mathsf{clen} \ll \delta^{20} n$.

   Output has high min-entropy Given $\hat{X}, \hat{Y}$ which are independent sources with min-entropy $\delta^2 n/100$ each, $\mathsf{Challenge}(\hat{X}, \hat{Y})$ is statistically close to having min-entropy $\Omega(\mathsf{clen})$.

   In extractor terminology, these conditions simply say that $\mathsf{Challenge}$ is a *condenser* for 2 independent sources. In [BKS+05] such a function (in fact a somewhere random extractor) was constructed using results from additive number theory.

2. A polynomial time computable function $\mathsf{Response} : \{0,1\}^n \times \{0,1\}^n \to (\{0,1\}^{\mathsf{clen}})^\ell$. We interpret the output as a list of ℓ matrices that have the same dimensions as the challenge matrix given by the $\mathsf{Challenge}$ function above. We use the somewhere extractors from Theorem 4.3 as the function $\mathsf{Response}$. Below we recall that this function satisfies the following properties (that will also be used in our final construction):

   - **Few matrices** $\ell = \mathrm{poly}(n)$[6].
   - **Hitting matrices** Given $\hat{X}, \hat{Y}$, independent sources with min-entropy $\delta^3 n$ each and any fixed matrix $\alpha \in \{0,1\}^{\mathsf{clen}}$, there there exists i and low deficiency subsources $X' \subset \hat{X}, Y' \subset \hat{Y}$ such that in these subsources $\mathsf{Response}(X', Y')_i = \alpha$ with probability 1.
   - **Fixed matrices on low deficiency subsources** Given any index i and any independent sources $\hat{X}, \hat{Y}$, we can decompose $(\hat{X}, \hat{Y})$ into a convex combination of low deficiency independent sources such that for every element of the combination X', Y', $\mathsf{Response}(X', Y')_i$ is fixed to a constant.

[6] In [BKS+05] the component they use for this step has an ℓ which is only constant. We can tolerate a much larger ℓ here because of the better components available to us.

Given the explicit functions $\mathsf{Challenge}$ and $\mathsf{Response}$ satisfying the properties above, we can now discuss how to use them to find the index j given samples $x \leftarrow_{\mathrm{R}} X$ and $y \leftarrow_{\mathrm{R}} Y$.

Definition 5.2. Given a challenge matrix and a list of response matrices, we say that the challenge is *responded* by the response if the challenge matrix is equal to one of the matrices in the response.

To find the index j:

1. Compute the response $\mathsf{Response}(x, y)$.
2. For every $i \in [t]$, compute a challenge $\mathsf{Challenge}(x_i, y)$.
3. Set r to be the smallest i for which $\mathsf{Challenge}(x_i, y)$ was *not responded* by $\mathsf{Response}(x, y)$.

We remind the reader that we will prove that the disperser works by arguing about *subsources* of the original adversarially chosen sources X, Y. Recall that we are currently working with the subsource $\hat{X} \subset X$ which has the properties guaranteed by Informal Lemma 5.1. Using the functions $\mathsf{Challenge}$ and $\mathsf{Response}$, we can then prove the following lemma:

Informal Lemma 5.3. *There exist low deficiency subsources $X^{\mathsf{good}} \subset \hat{X}, Y^{\mathsf{good}} \subset Y$ s.t. in these subsources $r = j$ with high probability.*

Proof Sketch: The lemma will follow from two observations.

Informal Claim 5.4. There are subsources $X^{\mathsf{good}} \subset \hat{X}, Y^{\mathsf{good}} \subset Y$ in which for every $i < j$, $\mathsf{Challenge}(X_i^{\mathsf{good}}, Y^{\mathsf{good}})$ is responded by $\mathsf{Response}(X^{\mathsf{good}}, Y^{\mathsf{good}})$ with probability 1. Furthermore X^{good} is a block-source (with roughly the same entropy as X) and Y^{good} has roughly the same entropy as Y.

Proof Sketch: Note that for $i < j$, $\hat{X}_i$ is fixed to a constant, so $\mathsf{Challenge}(\hat{X}_i, Y)$ is a function only of Y. Since the output length of $\mathsf{Challenge}$ is only clen bits, this implies (by Lemma 3.13) that there exists a subsource $\hat{Y} \subset Y$ of deficiency at most $\mathsf{clen} \cdot \mathsf{t}$ such that $\mathsf{Challenge}(\hat{X}_i, \hat{Y})$ is fixed for every $i < j$.

We can then use the **{Hitting matrices}** property of $\mathsf{Response}$ to find smaller subsources $X' \subset \hat{X}, Y' \subset Y$ s.t. there exists an index h_1 for which $\Pr[\mathsf{Challenge}(X_1', Y') = \mathsf{Response}(X', Y')_{h_i}] = 1$. Repeating this, we eventually get subsources $X^{\mathsf{good}} \subset \hat{X}, Y^{\mathsf{good}} \subset Y$ s.t. for every $i < j$, there exists an index h_i such that s.t. $\Pr[\mathsf{Challenge}(X_i^{\mathsf{good}}, Y^{\mathsf{good}}) = \mathsf{Response}(X^{\mathsf{good}}, Y^{\mathsf{good}})_{h_i}] = 1$, i.e., the challenge of every part of the source before the jth part is responded with probability 1 in these subsources.

The fact that X^{good} remains a block-source follows from Corollary 3.19. $\square$

Informal Claim 5.5. $\mathsf{Challenge}(X_j^{\mathsf{good}}, Y^{\mathsf{good}})$ is not responded by $\mathsf{Response}(X^{\mathsf{good}}, Y^{\mathsf{good}})$ with high probability.

Proof Sketch: The argument will use the union bound over ℓ events, one for each of the ℓ matrices in the response. We want to ensure that each matrix in the response is avoided by the challenge. Consider the ith matrix in the response $\mathsf{Response}(X^{\mathsf{good}}, Y^{\mathsf{good}})_i$. By the **{Fixed matrices on low deficiency subsources}** property of $\mathsf{Response}$, we know that $X^{\mathsf{good}}, Y^{\mathsf{good}}$ is a convex combination of independent sources in which the ith matrix is fixed to a constant. For every element of this convex combination, the probability that the challenge is equal to the ith response is extremely small by the property that the output of $\mathsf{Challenge}$ has high min-entropy. $\square$

Step 3: Computing the output of the disperser The output of the disperser is then just $\mathsf{BExt}(x_{[r]} \circ x, y)$. To show that our algorithm outputs a distribution with large support, first note that $\mathsf{BExt}(X_{[r]}^{\mathsf{good}} \circ X^{\mathsf{good}}, Y^{\mathsf{good}})$ is a subsource of $\mathsf{BExt}(X_{[r]} \circ X, Y)$. Thus it is sufficient to show that that $\mathsf{BExt}(X_{[r]}^{\mathsf{good}} \circ X^{\mathsf{good}}, Y^{\mathsf{good}})$ has a large support. However, by our choice of r, $r = j$ with high probability in $X^{\mathsf{good}}, Y^{\mathsf{good}}$. Thus $\mathsf{BExt}(X_{[r]}^{\mathsf{good}} \circ X^{\mathsf{good}}, Y^{\mathsf{good}})$ is statistically close to $\mathsf{BExt}(X_{[j]}^{\mathsf{good}} \circ X^{\mathsf{good}}, Y^{\mathsf{good}})$ and hence is statistically close to being uniform. $\square$

5.2 The Challenge-Response Mechanism in Our Application

Let us summarize how the challenge-response mechanism was used for linear min-entropy. The first step is to show that in any general source there is a small deficiency subsource which has some "nice structure". Intuitively, if the additional structure (in the last case the index j) was given to the construction, it would be easy to construct a disperser. The second step is to define a procedure (the challenge-response mechanism) which is able to "find" the additional structure with high probability, at least when run on some subsource of the good structured subsource. Thus, on the small subsource it is easy to construct a disperser. Since the disperser outputs two different values on the small subsource, it definitely does the same on the original source.

Now we discuss our disperser construction. In this discussion we will often be vague about the settings of parameters, but will give pointers into the actual proofs where things have been formalized.

There are several obstacles to adapting the challenge-response mechanism as used above to handle the case of min-entropy $k = n^{o(1)}$, which is what we achieve in this paper. Even the first step of the previos approach is problematic when the min-entropy k is less than $\sqrt{n}$. There we found a subsource of X which was block-source. Then we fixed the leading bits of the source to get a subsource which has a leading part which is fixed (no entropy), followed by a part with significant (medium) entropy, followed by the rest of the source which contains entropy even conditioned on the medium part.

When $k < \sqrt{n}$, on the one hand, to ensure that a single part of the source X_i cannot contain all the entropy of the source (which would make the above approach fail), we will have to make each part be smaller than $\sqrt{n}$ bits. On the other hand, to ensure that some part of the source contains at least one bit of min-entropy, we will have to ensure that there are at most $\sqrt{n}$ parts, otherwise our construction will fail for the situation in which each part of the source contains $k/\sqrt{n}$ bits of entropy. These two constraints clearly cannot be resolved simultaneously. Thus it seems like there is no simple deterministic way to partition the source in a way which nicely splits the entropy of the source.

The fix for this problem is to use recursion. We will consider parts of very large size (say $n^{0.9}$), so that the parts may contain all the entropy of the source. We will then develop a finer grained challenge-response mechanism that we can use to handle three levels of entropy differently: low, medium or high, for each part of the source. If we encounter a part of the source that has low entropy, as before we can fix it and ensure that our algorithm correctly identifies it as a block with low entropy. If we encounter a part which has a medium level of entropy, we can use the fact that this gives a way to partition the source into a block-source to produce a bit which is both 0 and 1 with positive probability. We will explain how we achieve this shortly. We note that here our situation is more complicated than [BKS+05] as we do not have an extractor that can work with a block-source with only two blocks for entropy below $\sqrt{n}$. Finally, if we encounter a part of the source which has a high entropy, then this part of the source is *condensed*, i.e. its entropy rate is significantly larger than that of the original source. Following previous works on seeded extractors, in this case we run the construction recursively on that part of the source (and the other source Y). The point is that we cannot continue these recursive calls indefinitely. After a certain number of such recursive calls, the source that we are working with will have to have such a high entropy rate that it *must* contain a part with a medium level of entropy.

Although this recursive description captures the intuition of our construction, to make the analysis of our algorithm cleaner, we open up the recursion to describe the construction and do the analysis.

Now let us give a more concrete description of our algorithm. Let $\mathsf{C}(\delta)$ be the number of blocks the extractor BExt of Theorem 4.1 requires for entropy $k = n^{\delta}$ and let t be some parameter to be specified later (think of t as a very small power of k).

We define a degree-t tree with depth $\log n / \log \mathsf{t} < \log n$ tree $\mathcal{T}_{n,\mathsf{t}}$ that we call the n, t *partition tree*. The nodes of $\mathcal{T}_{n,\mathsf{t}}$ are subintervals of $[1, n]$ defined in the following way:

1. The root of the tree is the interval $[1, n]$.
2. If a node v is identified with the interval $[a, b]$ of length greater than $k^{1/3}$, we let $v_1, \ldots, v_{\mathsf{t}}$ denote the t consecutive disjoint length-$|v|/\mathsf{t}$ subintervals of v. That is, $v_i = [a + \frac{b-a}{t}(i-1), a + \frac{b-a}{t}i]$. We let the i^{th} child of v be v_i.

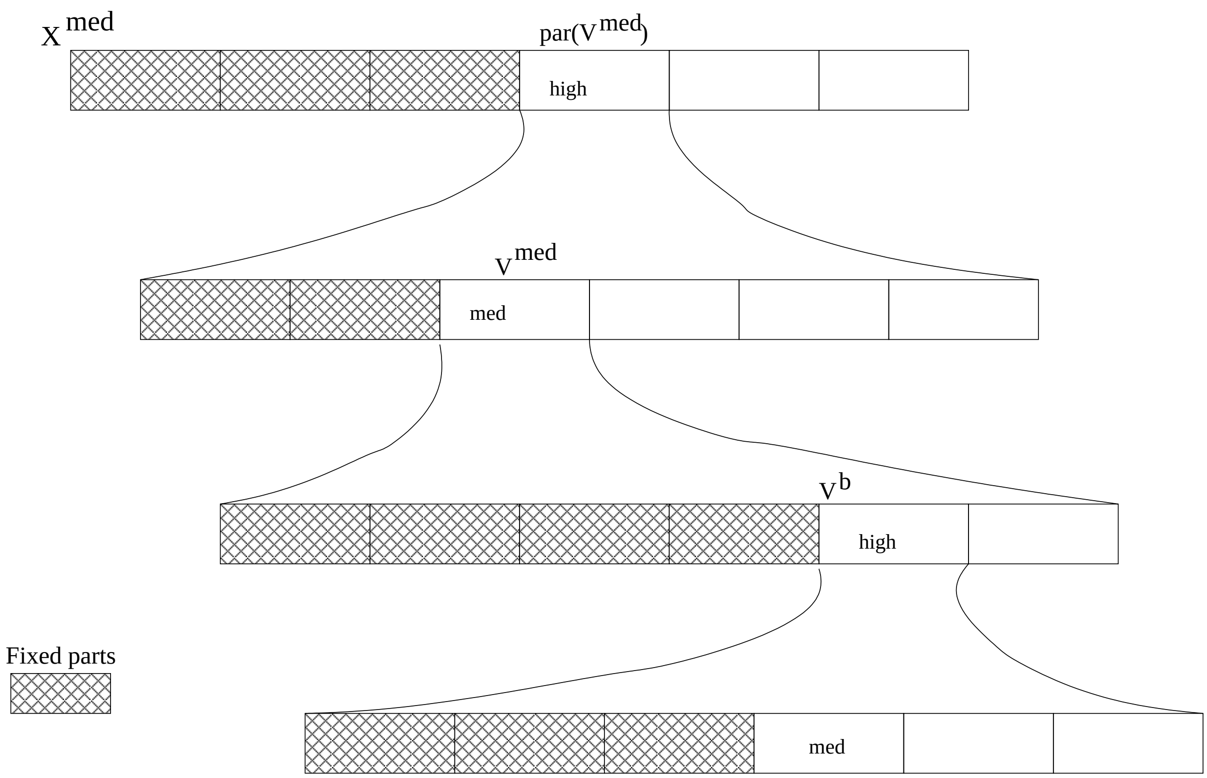

Figure 1: Finding two related medium parts in X^{med}

For a string $x \in \{0,1\}^n$ and a set $S \subseteq [1, n]$ we'll denote by x_S the projection of x onto the coordinates of S. If v is a node in $\mathcal{T}_{n,\mathsf{t}}$ then x_v denotes the projection of x onto the interval v.

Step 1 of analysis In analogy with our discussion for the case of linear min-entropy, we can show that any source X with min-entropy k contains a very nice structured low deficiency subsource $\hat{X}$. We will show that there is a vertex v^{b} in the tree s.t.:

- Every bit of $\hat{X}$ that precedes the bits in v^{b} is fixed.

- There are C children $i_1, \ldots, i_{\mathsf{C}}$ of v^{b} s.t. $\hat{X}_{i_1}, \hat{X}_{i_2}, \ldots, \hat{X}_{i_{\mathsf{C}}}$ is a C-block-source with entropy at least $\sqrt{k}$ in each block (even conditioned on previous blocks).
- There is an ancestor v^{med} of v^{b} such that $\hat{X}_{v^{\mathsf{med}}}, \hat{X}$ is a block-source with $k^{0.9}$ entropy in each block.

These three properties are captured in Figure 1.

This is done formally in **Step 1** of the analysis.

As in the case of linear min-entropy, we would be in great shape if we were given $v^{\mathsf{b}}, v^{\mathsf{med}}, i_1, \ldots, i_{\mathsf{C}}$. Of course we don't know these and even worse, this time we will not even be able to identify all of these with high probability in a subsource. Another obstacle to adapting the construction for linear min-entropy to the case of $k = n^{o(1)}$ is that we don't have a simple replacement for the function $\mathsf{Challenge}$ that we had for the case of linear min-entropy. However we will be able to use the components that are available to us to compute challenge matrices which are still useful.

The construction will proceed as follows:

1. For every vertex v of the tree, we will compute a small ($\mathsf{nrows} \times \mathsf{clen}$) challenge matrix $\mathsf{Challenge}(x_v, y)$ of size $\mathsf{len} = \mathsf{nrows} \cdot \mathsf{clen}$, that is a function only of the bits that correspond to that vertex in x and all of y.
2. For every vertex v of the tree, we will associate a response $\mathsf{Response}(x_v, y)$ which is interpreted as a list of $\mathrm{poly}(n)$ matrices each of size $\mathsf{len} = \mathsf{nrows} \cdot \mathsf{clen}$.

For every vertex v in the tree, we will call the set of vertices whose intervals lie strictly to the left of v (i.e. the interval does not intersect v and lies to the left of v), and whose parent is an ancestor of v, the *left family* of v. In **Step 2** of the formal analysis, we will find low deficiency subsources $X^{\mathsf{good}} \subset \hat{X}, Y^{\mathsf{good}} \subset Y$ s.t. for every vertex v which is in the left family of v^{b}, $\mathsf{Challenge}(X^{\mathsf{good}}_v, Y^{\mathsf{good}})$ is a fixed matrix that occurs in $\mathsf{Response}(X^{\mathsf{good}}_{\mathsf{par}(v)}, Y^{\mathsf{good}})$ with probability 1.

In **Step 3** of the formal analysis, we will show that for every vertex v which lies on the path from v^{b} to the root $\mathsf{Challenge}(X^{\mathsf{good}}_v, Y^{\mathsf{good}})$ is statistically close to being somewhere random. For technical reasons we will actually need a property which is stronger than this. We will actually show that for every vertex v which lies on the path from v^{b} to the root and *all low deficiency subsources* $X' \subset X^{\mathsf{good}}, Y' \subset Y^{\mathsf{good}}$, $\mathsf{Challenge}(X'_v, Y')$ is statistically close to being somewhere random.

At this point we will have made a lot of progress in the construction and analysis. We have found subsources $X^{\mathsf{good}}, Y^{\mathsf{good}}$ s.t. the challenges for all the vertices that occur to the left of the path to v^{b} have been fixed. Moreover the challenges for vertices on this good path have high min-entropy, even if we move to *any* subsources of small deficiency X', Y'. In some sense we will have identified the good path that goes to v^{b} in these subsources, though we still don't know where $v^{\mathsf{b}}, v^{\mathsf{med}}$ are on this path. From here we will need to do only a little more work to compute the output of the disperser.

Now let us describe how we compute the challenges and ensure the properties of $X^{\mathsf{good}}, Y^{\mathsf{good}}$ that we discussed above more concretely. We will need the following components:

1. To generate the challenges, we will need a polynomial time computable function $\mathsf{BExt} : (\{0,1\}^n)^{\mathsf{C}} \times \{0,1\}^n \to \{0,1\}^{\mathsf{clen}}$ that is an extractor for a $(\mathsf{C}, \sqrt{k})$ block-source and an independent $\sqrt{k}$ source. Here think of clen as roughly $k^{0.9}$.
2. The second component is exactly the same as the second component from the case of linear min-entropy and will be used to generate the responses. We need a polynomial time computable function $\mathsf{Response} : \{0,1\}^n \times \{0,1\}^n \to (\{0,1\}^{\mathsf{len}})^{\ell}$ (the output is interpretted as a list of ℓ $\mathsf{nrows} \times \mathsf{clen}$ matrices) with the property that:

- **Few outputs** $\ell = \mathrm{poly}(n)$.
- **Hitting matrices** Given $\hat{X}, \hat{Y}$, independent sources with min-entropy $\sqrt{k}$ each and any fixed $\mathsf{nrows} \times \mathsf{clen}$ matrix c, there there exists i and low deficiency subsources $X' \subset \hat{X}, Y' \subset \hat{Y}$ such that in these subsources $\mathsf{Response}(X', Y')_i = c$ with probability 1.
- **Fixed matrices on low deficiency subsources** Given any independent sources $\hat{X}, \hat{Y}$, and an index i, $(\hat{X}, \hat{Y})$ is a convex combination of low deficiency independent sources such that for every element (X', Y') of the combination, $\mathsf{Response}(X', Y')_i$ is fixed to a constant.

As before, we will use the function SE promised by Theorem 4.3 for this component.

We define for every node v of the tree a relatively small challenge matrix $\mathsf{Challenge}(x_v, y)$ with nrows rows of length clen each. We will set up the size of these challenge matrices as roughly $\mathsf{len} = k^{0.9}$.

Let $x_{v_1}, \ldots, x_{v_t}$ be the division of x_v to t sub-parts. Then, we let $\mathsf{Challenge}(x_v, y)$ contain one row that is equal to $\mathsf{BExt}(x_{v_{i_1}} \circ \ldots \circ x_{v_{i_{\mathsf{C}}}}, y)$ for every possible C-tuple $1 \le i_1 < i_2 < \cdots < i_{\mathsf{C}} \le t$. If v is a leaf then $\mathsf{Challenge}(x_v, y)$ has no other rows and we will pad the matrix with 0's to make it of size $\mathsf{nrows} \cdot \mathsf{clen}$. If v is a non-leaf then we let $\mathsf{Challenge}(x_{v_1}, y), \ldots, \mathsf{Challenge}(x_{v_t}, y)$ be the challenges of all the children of v in the tree. We will append the rows of $\mathsf{Challenge}(x_{v_i}, y)$ to $\mathsf{Challenge}(x_v, y)$ where i is the smallest index such that $\mathsf{Challenge}(x_{v_i}, y)$ does not equal any of the matrices in $\mathsf{Response}(x_v, y)$. Again, if the matrix we obtain contains fewer than nrows rows, we pad it with 0s to ensure that it is of the right size.

Note that in this way every challenge $\mathsf{Challenge}(x_v, y)$ is indeed only a function of the bits in x_v, y. This will be crucial for our analysis.

Step 2 of analysis: ensuring that challenges are responded in left family The following claim is proved in **Step 2** of the analysis (Claim 6.12).

Informal Claim 5.6 (Left family challenges are responded)**.** There are subsources $X^{\mathsf{good}} \subset \hat{X}, Y^{\mathsf{good}} \subset Y$ in which for every vertex w to the left of v^{b} whose parent $\mathsf{par}(w)$ lies on the path from v^{b} to the root, $\mathsf{Challenge}(X^{\mathsf{good}}_w, Y^{\mathsf{good}})$ is responded by $\mathsf{Response}(X^{\mathsf{good}}_{\mathsf{par}(w)}, Y^{\mathsf{good}})$ with probability 1.

Proof Sketch: Note that for w which is to the left of v^{b}, $\hat{X}_w$ is fixed to a constant, so $\mathsf{Challenge}(\hat{X}_w, Y)$ is a function only of Y. Since the output length of $\mathsf{Challenge}$ is only len bits, this implies (by Lemma 3.13) that there exists a subsource $\hat{Y} \subset Y$ of deficiency at most $\mathsf{len} \cdot \mathsf{t} \log n$ such that $\mathsf{Challenge}(\hat{X}_w, \hat{Y})$ is fixed for every such w. Then, since $\hat{X}_v, \hat{Y}$ are still high entropy sources for every v on the path from v^{b} to the root, we can repeatedly use the **{Hitting matrices}** property of $\mathsf{Response}$ to find smaller subsources $X^{\mathsf{good}} \subset \hat{X}, Y^{\mathsf{good}} \subset \hat{Y}$ s.t. for every w to the left of v, $\exists l$ s.t. $\Pr[\mathsf{Challenge}(\hat{X}_w, \hat{Y}) = \mathsf{Response}(\hat{X}_{\mathsf{par}(w)}, \hat{Y})_l] = 1$. □

Step 3 of analysis: ensuring that challenges along the good path are somewhere random We argue that the challenges along the good path are statistically close to being somewhere random in $X^{\mathsf{good}}, Y^{\mathsf{good}}$. This is done formally in **Step 3** in Lemma 6.13. The intuition for this is that first the challenge associated with the vertex v^{b} is somewhere random since v^{b} has children that form a block-source. We will then show that with high probability this challenge of v^{b} appears in the challenge matrix of every ancestor of v^{b}.

Informal Claim 5.7 (Challenges along path to v^{b} are somewhere random)**.** For all low deficiency subsources $X' \subset X^{\mathsf{good}}, Y' \subset Y^{\mathsf{good}}$ and any vertex v that's on the path from v^{b} to the root, $\mathsf{Challenge}(X'_v, Y')$ is statistically close to being somewhere random.

Proof Sketch: We will prove this by induction on the distance of the vertex v from v^{b} on the path. When $v = v^{\mathsf{b}}$, note that $\mathsf{Challenge}(X'_{v^{\mathsf{b}}}, Y')$ contains $\mathsf{BExt}(x_{v_{i_1}} \circ \ldots \circ x_{v_{i_u}}, y)$ for every C-tuple of children $v_{i_1}, \ldots, v_{i_{\mathsf{C}}}$ of v^{b}. By the guarantee on $\hat{X}$, we know that there exist $i_1, \ldots, i_{\mathsf{C}}$ s.t. $\hat{X}_{v_{i_1}}, \ldots, \hat{X}_{v_{i_{\mathsf{C}}}}$ is a C-block-source. Since X' is a low deficiency subsource of $\hat{X}$, $X'_{v_{i_1}}, \ldots, \hat{X}'_{v_{i_{\mathsf{C}}}}$ must also be close to a C block-source by Corollary 3.19. Thus we get that $\mathsf{Challenge}(X'_{v^{\mathsf{b}}}, Y')$ is statistically close to somewhere random.

To do the inductive step we show that $\mathsf{Challenge}(X'_{\mathsf{par}(v)}, Y')$ is close to being somewhere random given that $\mathsf{Challenge}(X''_v, Y'')$ is somewhere random for even smaller subsources $X'' \subset X', Y'' \subset Y'$.

The argument will use the union bound over ℓ events, one for each of the ℓ strings in the response. We want to ensure that each string in the response is avoided by the challenge. Consider the ith string in the response $\mathsf{Response}(X'_{\mathsf{par}(v)}, Y')_i$. By the **{Fixed matrices on low deficiency subsources}** property of $\mathsf{Response}$, we know that X', Y' is a convex combination of independent sources in which the ith string is fixed to a constant.

Now every element of this convex combination X'', Y'' is a subsource of the original sources, the probability that $\mathsf{Challenge}(X''_v, Y'')$ is equal to the ith response is extremely small by the property that the output of $\mathsf{Challenge}(X''_v, Y'')$ has high min-entropy. Thus with high probability $\mathsf{Challenge}(X'_{\mathsf{par}(v)}, Y')$ contains $\mathsf{Challenge}(X'_v, Y')$ as a substring. This implies that $\mathsf{Challenge}(X'_{\mathsf{par}(v)}, Y')$ is statistically close to being somewhere random. □

Step 4 of analysis: ensuring that the disperser outputs both 0 and 1 The output for our disperser is computed in a way that is very different from what was done for the case of linear min-entropy. The analyis above included two kinds of tricks:

- When we encountered a part of the source which had a low amount of entropy, we went to a subsource where the part was fixed and the corresponding challenge was responded with probability 1.
- When we encountered a part of the source which had a high level of entropy, we went to a subsource where the corresponding challenge is not responded with high probability

The intuition for our disperser is that if we encounter a part of source (such as v^{med} above) which both has high min-entropy and such that fixing that part of the source still leaves enough entropy in the rest of the source, we can ensure that the challenge is both responded and not responded with significant probability. We will elaborate on how to do this later on. This is very helpful as it gives us a way to output two different values! By outputting "0" in case the challenge is responded and "1" in case it is not we obtain a disperser. Now let us be more concrete.

Definition 5.8. Given two $\mathsf{nrows} \times \mathsf{clen}$ matrices and an integer $1 \leq q \leq \mathsf{clen}$, we say that one matrix is q-responded by the other if the first q columns of both matrices are equal.

The first observation is the following claim which is proved formally in **Step 4** (Lemma 6.14). The claim will be used with $q \ll \mathsf{clen}, \mathsf{len}$.

Below we use the symbol $\lesssim$ to denote an inequality that is only approximate in the sense that in the formal analysis there are small error terms (which may be ignored for the sake of intuition) that show up in the expressions.

Informal Claim 5.9. For every vertex v on the path from v^{b} to the root,

$$\Pr[\mathsf{Challenge}(X^{\mathsf{good}}_v, Y^{\mathsf{good}}) \text{ is } q\text{-responded by } \mathsf{Response}(X^{\mathsf{good}}_{\mathsf{par}(v)}, Y^{\mathsf{good}})] \lesssim 2^{-q}$$

Proof Sketch: As before, we will use the **{Fixed matrices on low deficiency subsources}** property of $\mathsf{Response}$ and the fact that $\mathsf{Challenge}(X'_v, Y')$ is somewhere random for any low deficiency subsources $X' \subset X^{\mathsf{good}}, Y' \subset Y^{\mathsf{good}}$ to argue that the probability that for every index q,

$$\Pr[\mathsf{Challenge}(X_v^{\mathsf{good}}, Y^{\mathsf{good}}) \text{ is } q\text{-responded by } \mathsf{Response}(X_{\mathsf{par}(v)}^{\mathsf{good}}, Y^{\mathsf{good}})_q] \lesssim 2^{-q}$$

Then we just apply a union bound over the poly(n) response strings to get the claim. □

Next we observe that for the vertex v^{med}, its challenge is reponded with a probability that behaves very nicely. In particular, note that we get that the challenge is both responded and not responded with noticeable probability. This is Lemma 6.16 in the formal analysis.

Informal Claim 5.10.

$$2^{-q\cdot\mathsf{nrows}} \lesssim \Pr[\mathsf{Challenge}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}}) \text{ is } q\text{-responded by } \mathsf{Response}(X_{\mathsf{par}(v^{\mathsf{med}})}^{\mathsf{good}}, Y^{\mathsf{good}})] \lesssim 2^{-q}$$

Proof Sketch: The idea is that $X_{\mathsf{par}(v^{\mathsf{med}})}^{\mathsf{good}}$ is a convex combination of sources $X'_{\mathsf{par}(v^{\mathsf{med}})}$ in which $X'_{\mathsf{par}(v^{\mathsf{med}})}$ is fixed, but X' still has a significant amount of entropy. Thus we are in the situation where we proved Claim 5.6. We can then show that X', Y^{good} are a convex combination of sources X'', Y'' s.t. $\mathsf{Challenge}(X''_{v^{\mathsf{med}}}, Y'')$ is fixed to a constant. Thus

$$\Pr[\mathsf{Challenge}(X''_{v^{\mathsf{med}}}, Y'') \text{ is } q\text{-responded by } \mathsf{Response}(X''_{\mathsf{par}(v^{\mathsf{med}})}, Y'') \gtrsim 2^{-q\cdot\mathsf{nrows}}$$

This implies that

$$\Pr[\mathsf{Challenge}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}}) \text{ is } q\text{-responded by } \mathsf{Response}(X_{\mathsf{par}(v^{\mathsf{med}})}^{\mathsf{good}}, Y^{\mathsf{good}}) \gtrsim 2^{-q\cdot\mathsf{nrows}}$$

The upper bound is just a special case of Claim 5.9. □

Given these two claims, here is how we define the output of the disperser:

1. We define a sequence of decreasing challenge lengths: $\mathsf{clen} \gg \mathsf{clen}_{1,0} \gg \mathsf{clen}_{1,1} \gg \mathsf{clen}_{1,2} \gg \mathsf{clen}_{2,0} \gg \mathsf{clen}_{2,1} \gg \mathsf{clen}_{2,2} \gg \mathsf{clen}_{3,0} \cdots$.
2. If v is not a leaf, let $v_1, \ldots, v_{\mathsf{t}}$ be v's t children. Let q be the depth of v. If for every i $\mathsf{Challenge}(x_{v_i}, y)$ is $\mathsf{clen}_{q,0}$-responded by $\mathsf{Response}(x_{v_i}, y)$, set $\mathsf{val}(x_v, y) = 0$, else let i_0 be the smallest i for which this doesn't happen. Then,
   (a) If $\mathsf{Challenge}(x_{v_{i_0}}, y)$ is $\mathsf{clen}_{q,1}$-responded by $\mathsf{Response}(x_v, y)$, set $\mathsf{val}(x_v, y) = 1$.
   (b) Else if $\mathsf{Challenge}(x_{v_{i_0}}, y)$ is $\mathsf{clen}_{q,2}$-responded but not $\mathsf{clen}_{q,1}$-responded by $\mathsf{Response}(x_v, y)$, set $\mathsf{val}(x_v, y) = 0$.
   (c) Else set $\mathsf{val}(x_v, y) = \mathsf{val}(x_{v_{i_0}}, y)$.
3. The disperser outputs $\mathsf{val}(x, y)$.

Let h be the depth of v^{med}. The correctness is then proved by proving two more claims:

Informal Claim 5.11. The probability that $\mathsf{val}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}})$ differs from $\mathsf{val}(X^{\mathsf{good}}, Y^{\mathsf{good}})$ is bounded by $2^{-\mathsf{clen}_{h,0}}$.

Proof Sketch: In fact we can argue that with high probability, $\mathsf{val}(X^{\mathsf{good}}_{v^{\mathsf{med}}}, Y^{\mathsf{good}}) = \mathsf{val}(X^{\mathsf{good}}_{\mathsf{par}(v^{\mathsf{med}})}, Y^{\mathsf{good}}) = \mathsf{val}(X^{\mathsf{good}}_{\mathsf{par}(\mathsf{par}(v^{\mathsf{med}}))}, Y^{\mathsf{good}}) = \cdots = \mathsf{val}(X^{\mathsf{good}}, Y^{\mathsf{good}})$. The reason is that by Claim 5.9, for any vertex v on the path from v^{med} to the root at depth q,

$$\Pr[\mathsf{val}(X^{\mathsf{good}}_v, Y^{\mathsf{good}}) \neq \mathsf{val}(X^{\mathsf{good}}_{\mathsf{par}(v)})] \lesssim 2^{-\mathsf{clen}_{q,0}} \ll 2^{-\mathsf{clen}_{h,0}}$$

Thus, by the union bound, we get that with high probability all of these are in fact equal. □

Next, we will argue that $\mathsf{val}(X^{\mathsf{good}}_{v^{\mathsf{med}}}, Y^{\mathsf{good}})$ is both 0 and 1 with significant probability. This will complete the proof, since this will show that $\mathsf{val}(X^{\mathsf{good}}, Y^{\mathsf{good}})$ is both 0 and 1 with significant probability.

Informal Claim 5.12.

$$\Pr[\mathsf{val}(X^{\mathsf{good}}_{v^{\mathsf{med}}}, Y^{\mathsf{good}})] = 1] \gtrsim 2^{-\mathsf{clen}_{h,1}}$$

$$\Pr[\mathsf{val}(X^{\mathsf{good}}_{v^{\mathsf{med}}}, Y^{\mathsf{good}})] = 0] \gtrsim 2^{-\mathsf{clen}_{h,2}}$$

Proof Sketch: This follows from Claim 5.10. The probability that $\mathsf{val}(X^{\mathsf{good}}_{v^{\mathsf{med}}}, Y^{\mathsf{good}}) = 1$ is lowerbounded by the probability that $\mathsf{Challenge}(X^{\mathsf{good}}_{v^{\mathsf{med}}}, Y^{\mathsf{good}})$ is $\mathsf{clen}_{h,1}$-responded by $\mathsf{Response}(X^{\mathsf{good}}_{\mathsf{par}(v^{\mathsf{med}})}, Y^{\mathsf{good}})$ minus the probability that $\mathsf{Challenge}(X^{\mathsf{good}}_{v^{\mathsf{med}}}, Y^{\mathsf{good}})$ is $\mathsf{clen}_{h,0}$-responded by $\mathsf{Response}(X^{\mathsf{good}}_{\mathsf{par}(v^{\mathsf{med}})}, Y^{\mathsf{good}})$. By Claim 5.10, we can ensure that this difference is significantly large.

The argument for the 0 output is very similar. □

These two claims then ensure that overall $\Pr[\mathsf{val}(X^{\mathsf{good}}, Y^{\mathsf{good}}) = 1] \gtrsim 2^{-\mathsf{clen}_{h,1}}$ and $\Pr[\mathsf{val}(X^{\mathsf{good}}, Y^{\mathsf{good}}) = 0] \gtrsim 2^{-\mathsf{clen}_{h,2}}$.

Thus $\mathsf{val}(X, Y) = \{0, 1\}$ as required.

6 Construction and analysis of the disperser

In this section we present the construction of the new 2-source disperser. We also prove that this construction works (thus proving our main theorems Theorem 1.10,Theorem 1.3 and Corollary 1.4. The formal presentation below closely follows the informal overview in Section 5.

6.1 Parameters

Setting the parameters We will first list the various parameters involved in the construction and say how we will set them.

- Let n be the length of the samples from the sources.
- Let k be the entropy of the input sources. Set $k = 2^{\log^{0.9} n}$.
- Let c_1 be the error constant from Theorem 4.1.
- Let $\mathsf{C} = O\left(\frac{\log n}{\log k}\right)$ be the number of blocks that the extractor BExt of Theorem 4.1 requires to extract from $\sqrt{k}$ entropy. (See Corollary 6.2 below for the precise parameters we use BExt for.) Without loss of generality, we assume that $c_1 \gg 1/\mathsf{C}$.
- We use t to denote the branching factor of the tree. We set $\mathsf{t} = n^{1/\mathsf{C}^4}$.
- We use $\mathsf{nrows} = \mathsf{t}^{\mathsf{C}} \log n$ to denote the maximum number of rows in any challenge matrix.

Name	Description	Restrictions	Notes
n	Input length		
k	Entropy	k	Assume $k \geq 2^{\log^{0.9} n}$
C	Number of blocks for BExt	$O(\log n / \log k)$	We always invoke BExt with entropy $\geq \sqrt{k}$
t	Degree of partition tree	$\mathsf{t} = n^{1/\mathsf{C}^4}$	
c_1	Error parameter of BExt		Inherited from BExt Corollary 6.2.
nrows	No. of rows in challenges and responses	$\mathsf{nrows} \leq (\log n)\mathsf{t}^{\mathsf{C}}$	
clen	Length of each row in challenges and responses	$\mathsf{clen} = n^{1/\mathsf{C}^2}$	
$\mathsf{clen}_{q,r}$	Shorter challenge lengths	$\mathsf{clen}_{q,r} = n^{\frac{1}{(3q+r)\mathsf{C}^2}}$	

Table 1: Parameters used in the construction

- We use clen to denote the length of every row in a challenge matrix. We set $\mathsf{clen} = n^{1/\mathsf{C}^2}$.
- We use $\mathsf{len} = \mathsf{nrows} \cdot \mathsf{clen}$ to denote the total size of the challenge matrices.
- We use $\mathsf{clen}_{q,r}$ to denote smaller challenge lengths and analogously define $\mathsf{len}_{q,r} = \mathsf{nrows} \cdot \mathsf{clen}_{q,r}$. We set $\mathsf{clen}_{q,r} = n^{\frac{1}{(3q+r)\mathsf{C}^2}}$.

Constraints needed in analysis Here are the constraints that the above parameters need to satisfy in the analysis.

- $\mathsf{t}^{1/\mathsf{C}^4} \geq 20\mathsf{C}$, used in the proofs of Lemma 6.9 and Lemma 6.10.
- $\frac{k}{(10\mathsf{t}^2 \cdot \mathsf{C})^2} \geq k^{0.9}$, used at the end of Step 1 in the analysis.
- $\mathsf{clen}^3 = o(k^{0.9})$, use at the end of Step 1 in the analysis and in the proof of Lemma 6.13.
- $\mathsf{clen} = o(k^{c_1})$, used in the proof of Lemma 6.15.
- $\mathsf{t} \cdot \mathsf{len} \cdot \log n = o(\mathsf{clen}^{2.1}) \Leftrightarrow \mathsf{t}^{\mathsf{C}+1} \cdot \log^2 n = o(\mathsf{clen}^{1.1})$, used at the end of Step 2 in the analysis.
- For any positive integers q, r, $\mathsf{nrows} = o(\mathsf{clen}_{q,r}/\mathsf{clen}_{q,r+1})$ and $\mathsf{nrows} = o(\mathsf{clen}_{q,r+2}/\mathsf{clen}_{q+1,r})$, used in the proof of Lemma 6.17.

6.2 Formal Construction

Definition 6.1. Given a challenge string $\mathsf{Challenge}$ interpretted as a $d \times \mathsf{len}$ boolean matrix with $d \leq \mathsf{nrows}$, a response string $\mathsf{Response}$ interpretted as a $\mathsf{nrows} \times \mathsf{len}$ boolean matrix, and a parameter q, we say that $\mathsf{Challenge}$ is *q-responded* by $\mathsf{Response}$, if the $d \times q$ sub-matrix of $\mathsf{Challenge}$ obtained by taking the first q bits from each row is equal to the $d \times q$ sub-matrix of $\mathsf{Response}$ obtained by taking the first q bits each from the first d rows of $\mathsf{Response}$.

6.2.1 Components

Block extractor We'll use the following corollary of Theorem 4.1:

Corollary 6.2 (Block Extractor). *There is a constant c_1 and a polynomial-time computable function* $\mathsf{BExt} : \{0,1\}^{\mathsf{C}n} \times \{0,1\}^n \to \{0,1\}^{\mathsf{out}}$ *s.t. if the parameters* C, n, k *are as above,*

For every every independent sources $X \in \{0,1\}^{\mathsf{C}n}$ *and* $Y \in \{0,1\}^n$ *with* $H_\infty(Y) \geq \sqrt{k}$ *and* $X = X_1 \circ \cdots \circ X_{\mathsf{C}}$ *a* $\sqrt{k}$ *block-source,*[7]

$$\left|\mathsf{BExt}(X,Y) - U_{\mathsf{clen}}\right| < 2^{-k^{c_1}}$$

Somewhere extractor with small error We will use the following corollary of Theorem 4.3 to generate our responses. We will set up SE to work on strings with entropy $\sqrt{k}$ with output length clen. For every string x of length at most n (if the input is shorter we will pad it to make it long enough), string $y \in \{0,1\}^n$, we define $\mathsf{Response}(x,y)$ to be the list of strings obtained from $\mathsf{SE}(x,y)$, by interpretting each row of the output of $\mathsf{SE}(x,y)$ as an $\mathsf{nrows} \times \mathsf{clen}$ boolean matrix.

Corollary 6.3 (Somewhere Extractor to generate Responses). *For every* n, k, len *that satisfy the constraints above, there is a polynomial time computable function* $\mathsf{Response} : (\{0,1\}^n)^2 \to (\{0,1\}^{\mathsf{len}})^\ell$ *(here the output is interpretted as a* $\mathsf{nrows} \times \mathsf{clen}$ *matrix) with the property that for any two* $(n, \sqrt{k})$ *sources* X, Y*,*

- ***Few outputs*** $\ell = \mathrm{poly}(n)$.
- ***Small error*** $\mathsf{Response}(X,Y)$ *is* $2^{-10\mathsf{len}}$*-close to a convex combination of somewhere random distributions and this property is strong with respect to both* X *and* Y*. Formally:*

$$\Pr_{y \leftarrow_R Y}[\mathsf{Response}(X,y) \textit{ is } 2^{-10\mathsf{len}}\textit{-close to being SR }] > 1 - 2^{-10\mathsf{len}}$$

- ***Hitting matrices*** *Let c be any fixed* $\mathsf{nrows} \times \mathsf{clen}$ *matrix. Then there are deficiency* $2\mathsf{len}$ *subsources* $\hat{X} \subset X, \hat{Y} \subset Y$ *such that* $\Pr[c \in \mathsf{SE}(\hat{X},\hat{Y})] = 1$.
- ***Fixed matrices on low deficiency subsources*** *Given any particular index i, there are* $20\mathsf{len}$ *deficiency subsources* $\hat{X} \subset X, \hat{Y} \subset Y$ *such that* $\mathsf{Response}(\hat{X},\hat{Y})_i$ *is a fixed matrix. Further,* X, Y *is* $2^{-10\mathsf{len}}$*-close to a convex combination of subsources such that for every* $\hat{X}, \hat{Y}$ *in the combination,*
  - $\hat{X}, \hat{Y}$ *are independent.*
  - $\mathsf{Response}(\hat{X},\hat{Y})_i$ *is constant.*
  - $\hat{X}, \hat{Y}$ *are of deficiency at most* $20\mathsf{len}$.

6.2.2 The Tree of Parts

We define a degree-t with depth $\log n / \log \mathsf{t} < \log n$ tree $\mathcal{T}_{n,\mathsf{t}}$ that we call the n, t *partition tree*. The nodes of $\mathcal{T}_{n,\mathsf{t}}$ are subintervals of $[1,n]$ defined in the following way:

1. The root of the tree is the interval $[1,n]$.

[7] That is, for every $i < \mathsf{C}$ and $x_1, \ldots, x_i \in \mathsf{Supp}(X_{1,\ldots,i})$, $H_\infty(X_{i+1}|x_1,\ldots,x_i) > 10\mathsf{clen}^5$.

2. If a node v is identified with the interval $[a, b]$ of length greater than $k^{1/3}$, we let $v_1, \ldots, v_{\mathsf{t}}$ denote the t consecutive disjoint length-$|v|/\mathsf{t}$ subintervals of v. That is, $v_i = [a + \frac{b-a}{t}(i-1), a + \frac{b-a}{t}i]$. We let the i^{th} child of v be v_i.

For a string $x \in \{0,1\}^n$ and a set $S \subseteq [1, n]$ we'll denote by x_S the projection of x onto the coordinates of S. If v is a node in $\mathcal{T}_{n,\mathsf{t}}$ then x_v denotes the projection of x onto the interval v.

6.2.3 Operation of the algorithm Disp

Algorithm 6.4.
$\mathsf{Disp}(x, y)$
Inputs: $x, y \in \{0,1\}^n$, Output: 1 bit.

1. On inputs $x, y \in \{0,1\}^n$, the algorithm Disp, working from the leaves upwards, will define for each node v in the tree $\mathcal{T}_{n,\mathsf{t}}$ a boolean challenge matrix ($\mathsf{Challenge}(x_v, y)$) with at most nrows rows, each of length clen in the following way:
   (a) If v is a leaf then $\mathsf{Challenge}(x_v, y)$ is the matrix with a single all 0s row.
   (b) If v is not a leaf then $\mathsf{Challenge}(x_v, y)$ is computed as follows:
      i. For each C-tuple $1 \leq i_1 < i_2 < \cdots < i_{\mathsf{C}} \leq \mathsf{t}$ let $S = v_{i_1} \cup v_{i_2} \cup \cdots \cup v_{i_{\mathsf{C}}}$ and append the row $\mathsf{BExt}(x_S, y)$ to the matrix $\mathsf{Challenge}(x_v, y)$.
      ii. Let $v_1, \ldots, v_{\mathsf{t}}$ be v's t children. If there exists an i such that $\mathsf{Challenge}(x_{v_i}, y)$ is not clen-responded by $\mathsf{Response}(x_v, y)$, let i_0 be the smallest such i and append all the rows of $\mathsf{Challenge}(x_{v_{i_0}}, y)$ to $\mathsf{Challenge}(x_v, y)$.
2. Next Disp will make a second pass on the tree, again working from the leaves upwards. This time it will define for each node v in the tree $\mathcal{T}_{n,\mathsf{t}}$ a bit $\mathsf{val}(x_v, y)$ in the following way:
   (a) If v is a leaf then $\mathsf{val}(x_v, y) = 0$.
   (b) If v is not a leaf, let $v_1, \ldots, v_{\mathsf{t}}$ be v's t children. Let q be the depth of v. If for every i $\mathsf{Challenge}(x_{v_i}, y)$ is $\mathsf{clen}_{q,0}$-responded by $\mathsf{Response}(x_{v_i}, y)$, set $\mathsf{val}(x_v, y) = 0$, else let i_0 be the smallest i for which this doesn't happen. Then,
      i. If $\mathsf{Challenge}(x_{v_{i_0}}, y)$ is $\mathsf{clen}_{q,1}$-responded by $\mathsf{Response}(x_v, y)$, set $\mathsf{val}(x_v, y) = 1$.
      ii. Else if $\mathsf{Challenge}(x_{v_{i_0}}, y)$ is $\mathsf{clen}_{q,2}$-responded but not $\mathsf{clen}_{q,1}$-responded by $\mathsf{Response}(x_v, y)$, set $\mathsf{val}(x_v, y) = 0$.
      iii. Else set $\mathsf{val}(x_v, y) = \mathsf{val}(x_{v_{i_0}}, y)$.
3. The output of Disp is $\mathsf{val}(x_{[1,n]}, y)$.

6.3 Formal Analysis

We now prove Theorem 1.10 which is the main Theorem of this paper. We need to prove that Disp is a 2-source disperser for min-entropy $k = 2^{\log^{0.9} n}$. and error parameter $\epsilon < 1/2$. We show that given two independent k-sources X and Y over n bits, $\mathsf{Disp}(X, Y)$ outputs both zero and one.

The analysis proceeds in several steps. In each step we make a restriction on one or both of the input sources. When we're done, we'll get the desired subsources X^{good}, Y^{good}.

Definition 6.5 (Path to a vertex)**.** Given a partition tree $\mathcal{T}_{n,\mathsf{t}}$ and a vertex v, let $\mathcal{P}_v$ to denote the path from the vertex v to the root in the tree $\mathcal{T}_{n,\mathsf{t}}$. That is, the set of nodes (including v) on the path from v to the root.

Definition 6.6 (Parent of a vertex)**.** Given a partition tree $\mathcal{T}_{n,\mathsf{t}}$ and a vertex v, let $\mathsf{par}(v)$ denote the parent of v.

Definition 6.7 (Left family of v)**.** Given a partition tree $\mathcal{T}_{n,\mathsf{t}}$ and a vertex v, let $\mathcal{L}_v$ denote the *left family* of v, i.e. if v is the interval $[c, d]$, define $\mathcal{L}_v = \{[a, b] \in \mathcal{T}_{n,\mathsf{t}} : a \leq c \text{ and } \mathsf{par}(w) \in \mathcal{P}_v\}$.

Note that for every vertex v, $|\mathcal{L}_v| = O(\mathsf{t} \log n)$, since the number of vertices in $\mathcal{P}_v$ is at most $\log n$.

6.3.1 Step 1: Preprocess X

The first step involves only the first source X. We'll restrict X to a subsource X^{med} that will have some attractive properties for us: we will ensure that in X^{med} there are a couple of parts which have entropy but do not have all the entropy of the source. We first prove a general lemma — Lemma 6.8 — and then use it to prove Lemma 6.9 and Lemma 6.10 to show that we obtain the desired subsource X^{med}.

Lemma 6.8 (Two-types lemma.)**.** *Let X be a general k source over $\{0,1\}^n$ divided into t parts $X = X_1 \circ \cdots \circ X_{\mathsf{t}}$. Let C be some positive integer and let $k' < k$ be such that $(\mathsf{C}+1)k' + 4t^2 \leq k$. Then, there exists a subsource $X' \subseteq X$ of deficiency at most $d = \mathsf{C}k' + 2\mathsf{t}^2$ that satisfies one of the following properties:*
Either

Somewhere high source — one high part *There exists $i \in [\mathsf{t}]$ such that the first $i-1$ parts of X' (namely $X'_1, \ldots, X'_{i-1}$) are* constant, *and $H_\infty(X'_i) \geq k'$.*

or

Somewhere block-source — C medium parts *There exist $0 < i_1 < i_2 < \cdots < i_{\mathsf{C}} \leq \mathsf{t}$ such that the first $i_1 - 1$ parts of X' are constant for every $j \in [\mathsf{C}]$, and $X'_{i_1}, X'_{i_2}, \ldots, X'_{i_{\mathsf{C}}}$ is a $(C, k'/t)$ block-source.*

Proof. We let $\tau_1 = 0$, $\tau_2 = k'/\mathsf{t}$, $\tau_3 = k'$ and $\tau_4 = n$ and use Lemma 3.20 to reduce X to a deficiency $2\mathsf{t}^2$ source X'' such that for every $i \in [\mathsf{t}]$ and every $x_1, \ldots, x_{i-1} \in \mathsf{Supp}(X''_{1,\ldots,i-1})$, the conditional entropy $H_\infty(X''_i | x_1, \ldots, x_{i-1})$ always falls into the same interval of $[0, k'/\mathsf{t}]$, $[k'/\mathsf{t}, k']$ and $[k', n]$ regardless of the choice $x_1, \ldots, x_i$.

We call parts where this conditional entropy falls into the interval $[0, k'/\mathsf{t})$ low, parts where this entropy falls into the interval $[k'/\mathsf{t}, k')$ medium and parts where it is at least k' high. We divide to two cases:

Case 1: if there are at most $\mathsf{C}-1$ medium parts before the first high part, we let i be the position of the first high part and fix the first $i-1$ parts to their most typical values. The conditional entropy X_1 given this prefix is still at least k'. Furthermore, since we fixed at most t low parts and at most C medium parts the overall deficiency is at most $(\mathsf{C}-1)k' + \mathsf{t}k'/\mathsf{t} = \mathsf{C}k'$.

Case 2: If there are at least C medium parts in the source, we let i be the position of the first medium part and fix the first $i-1$ parts to their most typical value. All medium parts remain medium conditioned on this prefix and the entropy we lose is at most $\mathsf{t}k'/\mathsf{t} \leq k'$.

□

We'll now use Lemma 6.8 to show that we can restrict the input source X to a subsource X^{sb} (for "somewhere block") satisfying some attractive properties:

Lemma 6.9. *Let X be a source over $\{0,1\}^n$ with min-entropy k. Let C, t be values satisfying $\mathsf{t}^{1/\mathsf{C}^4} \geq 20\mathsf{C}$. Then, there exists a deficiency $k/10 + 4\mathsf{t}^2 \log n$ subsource X^{sb} of X and a vertex v^{med} of $\mathcal{T}_{n,\mathsf{t}}$ with the following properties:*

- *For every $v \in \mathcal{L}_{v^{\mathsf{med}}}$, X_v^{sb} is fixed to a constant.*
- *The source $X^{\mathsf{sb}}_{\mathsf{par}(v^{\mathsf{med}})}$ is a $(\mathsf{C}, \frac{k}{20\mathsf{t}\mathsf{C}n^{1/\mathsf{C}^4}})$-somewhere block-source.*
- *$X^{\mathsf{sb}}_{v^{\mathsf{med}}}$ is the first block of the block-source in $X^{\mathsf{sb}}_{\mathsf{par}(v^{\mathsf{med}})}$.*

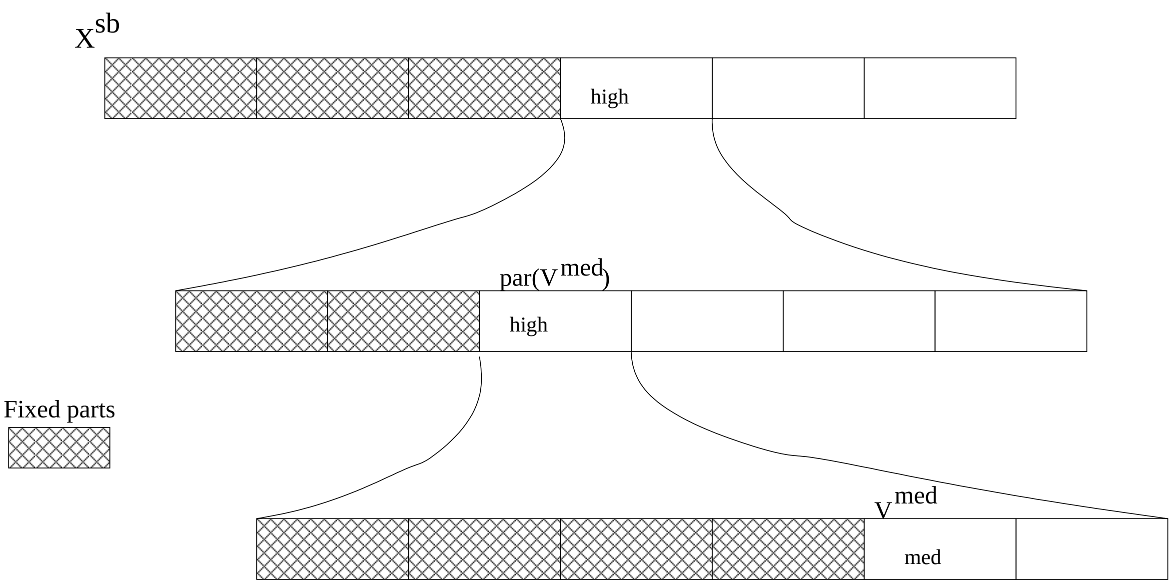

Figure 2: Finding a medium part in X^{sb}

Proof. We prove the lemma by induction on $\lceil \log(n/k) \rceil = \lceil \log n - \log k \rceil$. If $n = k$ then this is the uniform distribution and everything is trivial. We invoke Lemma 6.8 with parameter $k' = k/(20\mathsf{C})$ to obtain a deficiency $k/20 + 4\mathsf{t}^2$ subsource X' that is either k'-somewhere high or $(\mathsf{C}, k'/\mathsf{t})$-somewhere block-source.

If X' is $(\mathsf{C}, k'/\mathsf{t})$-somewhere block-source then set $X^{\mathsf{sb}} = X'$, $\mathsf{par}(v^{\mathsf{med}}) = [1, n]$ and v^{med} corresponding to the first part of the block-source given by Lemma 6.8. Since $k'/\mathsf{t} = k/(20\mathsf{t}\mathsf{C})$ we have that $X^{\mathsf{sb}}, v^{\mathsf{med}}$ satisfy the properties in the conclusion of the lemma.

The second possibility is that X' is a k'-somewhere high source. We let i be the index of the high block of entropy k', and let v_i be the corresponding interval. Note that X'_{v_j} attains some fixed value with probability 1, for all $j < i$. Let $n' = |v_i| = n/\mathsf{t}$. Since $\frac{n'}{k'} = \frac{n}{k}\frac{20\mathsf{C}}{\mathsf{t}} < \frac{n}{4k}$ we have that $\log(n'/k') < \log(n/k) - 2$ and so can assume by the induction hypothesis that the statement holds for the source $Z = X'_{v_i}$. This means that we have a subsource $Z' \subset Z$ of deficiency $k'/10 + 4\mathsf{t}^2 \log n'$ of Z and a node $\mathsf{par}(v^{\mathsf{med}})$ in the tree $\mathcal{T}_{n',\mathsf{t}}$ such that (below we use that $\mathsf{t}^{1/\mathsf{C}^4} \geq 20\mathsf{C}$):

- For every $v \in \mathcal{L}_{v^{\mathsf{med}}}$, Z'_v is fixed to a constant.
- The source $Z'_{\mathsf{par}(v^{\mathsf{med}})}$ is a $(\mathsf{C}, \frac{k'}{20\mathsf{tC}n'^{1/\mathsf{C}^4}} = \frac{k}{20\mathsf{tC}n^{1/\mathsf{C}^4}} \cdot \frac{\mathsf{t}^{1/\mathsf{C}^4}}{20\mathsf{C}} \geq \frac{k}{20\mathsf{tC}n^{1/\mathsf{C}^4}})$-somewhere block-source.
- $Z'_{v^{\mathsf{med}}}$ is the first block of the block-source in $Z'_{\mathsf{par}(v^{\mathsf{med}})}$.

We define X^{sb} to be the natural extension of the subsource Z' to a subsource of X'. Then we see that $X^{\mathsf{sb}} \subset X'$ is of deficiency at most $k'/10 + 4\mathsf{t}^2 \log n'$. Since $\log n' \leq \log n - 1$ and $k'/10 < k/20$, $k'/10 + 4\mathsf{t}^2 \log n' \leq k/20 + 4\mathsf{t}^2(\log n - 1)$. Hence $X^{\mathsf{sb}} \subset X$ is a source of deficiency at most $k/10 + 4\mathsf{t}^2 \log n$. It is clear that X^{sb} and $\mathsf{par}(v^{\mathsf{med}})$ satisfy our requirements. $\square$

Note that by our setting of parameters, the entropy of the medium part promised by the above lemma is actually $\frac{k}{20\mathsf{tC}n^{1/\mathsf{C}^4}} = \frac{k}{20\mathsf{t}^2\mathsf{C}}$.

Next we show that by invoking the above lemma twice, we can move to a subsource X^{med} that has even more structure.

Lemma 6.10. *Let X be a source over $\{0,1\}^n$ with min-entropy k. Let C, t be as above. Then, there exists a deficiency $k/5 + 8\mathsf{t}^2 \log n$ subsource X^{med} of X and three vertices $\mathsf{par}(v^{\mathsf{med}})$, v^{med} and $v^{\mathsf{b}} = [a,b]$ of $\mathcal{T}_{n,\mathsf{t}}$ with the following properties:*

- *v^{med} is an ancestor of v^{b}.*
- *The source $X^{\mathsf{med}}_{\mathsf{par}(v^{\mathsf{med}})}$ is a $(\mathsf{C}, \frac{k}{40\mathsf{tC}n^{1/\mathsf{C}^4}})$-somewhere block-source, and $X^{\mathsf{med}}_{v^{\mathsf{med}}}$ is the first medium block in this source.*
- *The source $X^{\mathsf{med}}_{v^{\mathsf{b}}}$ is a $(\mathsf{C}, \frac{k}{(20\mathsf{tC}n^{1/\mathsf{C}^4})^2})$-somewhere block-source.*
- *There is a value $x \in \{0,1\}^{a-1}$ such that $X^{\mathsf{med}}_{[1,a-1]} = x$ with probability 1.*

Proof. We prove this lemma by invoking Lemma 6.9 twice. We start with our source X and invoke Lemma 6.9 to find a subsource X^{sb} and vertices $\mathsf{par}(v^{\mathsf{med}}), v^{\mathsf{med}}$ as in the conclusion of the lemma. Next we apply the lemma again to $X^{\mathsf{sb}}_{v^{\mathsf{med}}}$.

Since $X^{\mathsf{sb}}_{v^{\mathsf{med}}}$ is a source on $n' < n$ bits with min-entropy $\frac{k}{20\mathsf{tC}n^{1/\mathsf{C}^4}}$, we get that there is a subsource $X^{\mathsf{med}} \subset X^{\mathsf{sb}}$ with deficiency at most $\frac{k}{400\mathsf{tC}n^{1/\mathsf{C}^4}} + 4\mathsf{t}^2 \log n$ and a vertex v^{b} which is a somewhere block-source. Since $X^{\mathsf{sb}} \subset X$ was of deficiency at most $k/10 + 4\mathsf{t}^2 \log n$, we get that $X^{\mathsf{med}} \subset X$ is a subsource of X with deficiency at most $k/5 + 8\mathsf{t}^2 \log n$. Further note that $H_\infty(X^{\mathsf{med}}_{v^{\mathsf{med}}}) \geq \frac{k}{20\mathsf{tC}n^{1/\mathsf{C}^4}} - \frac{k}{400\mathsf{tC}n^{1/\mathsf{C}^4}} - 4\mathsf{t}^2 \log n \geq \frac{k}{30\mathsf{tC}n^{1/\mathsf{C}^4}} - 4\mathsf{t}^2 \log n \geq \frac{k}{40\mathsf{tC}n^{1/\mathsf{C}^4}}$ by our choice of parameters. $\square$

We apply Lemma 6.10 to the input source X with our parameters k, t as chosen in Section 6.1. We obtain a deficiency $k/4$ subsource (since $4\mathsf{t}^2 = o(k)$) X^{med} of X, and three nodes $\mathsf{par}(v^{\mathsf{med}}), v^{\mathsf{med}}, v^{\mathsf{b}} = [a,b]$ in the tree $\mathcal{T}_{n,\mathsf{t}}$ satisfying (by our choice of parameters):

Result of Step 1: A deficiency $k/4$ subsource $X^{\mathsf{med}} \subset X$ satisfying:

v^{med} **is the leading block in a block-source:** $X^{\mathsf{med}}_{\mathsf{par}(v^{\mathsf{med}})}$ is a $(\mathsf{C}, \frac{k}{40\mathsf{tC}n^{1/\mathsf{C}^4}} \geq k^{0.9})$-somewhere block-source, with a sub-block $X^{\mathsf{med}}_{v^{\mathsf{med}}}$ which is the first non-constant "good" sub-block.

$X^{\mathsf{med}}_{v^{\mathsf{b}}}$ **has a block-source:** The source $X^{\mathsf{med}}_{v^{\mathsf{b}}}$ is a $(\mathsf{C}, \frac{k}{(10\mathsf{t}^2\mathsf{C})^2} \geq k^{0.9})$-somewhere block-source.

Fixed left family: For every $w \in \mathcal{L}_{v^{\mathsf{b}}}$ (Definition 6.7), X^{med}_w is fixed.

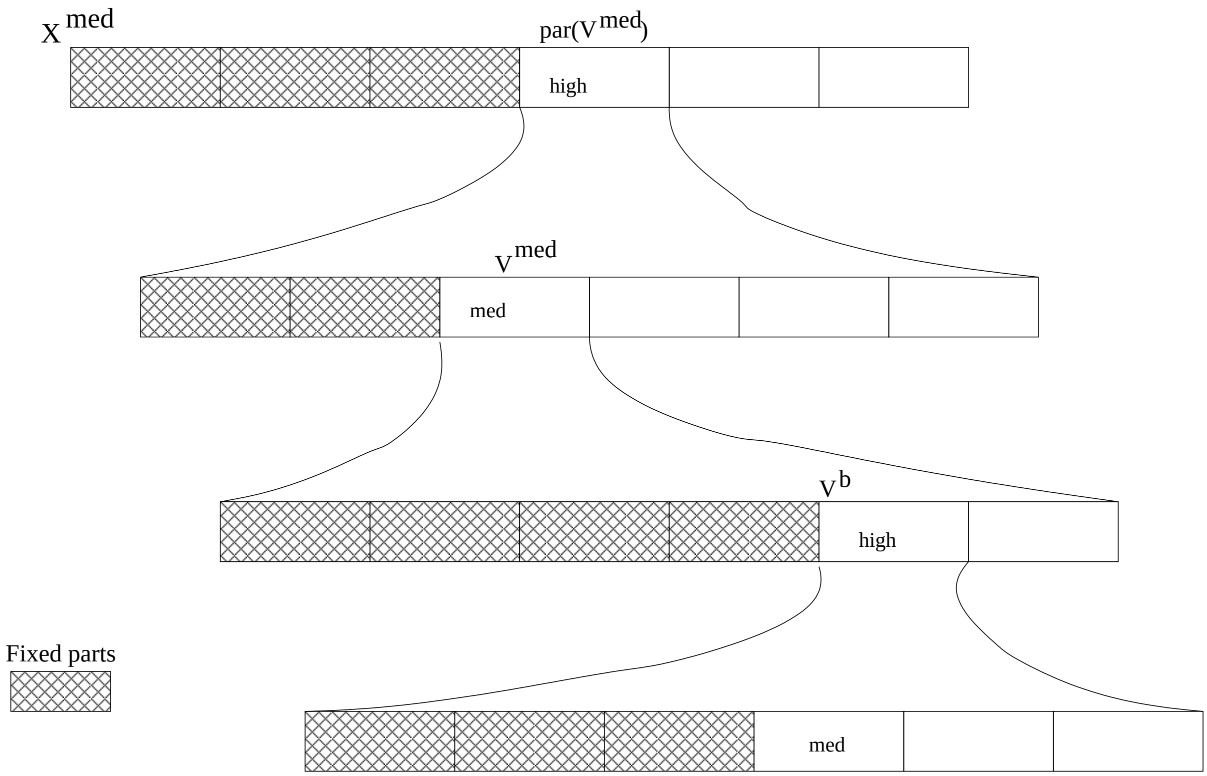

Figure 3: Finding two related medium parts in X^{med}

6.3.2 Step 2: Ensuring that challenges from the left family are properly responded.

Our desired good subsources X^{good} and Y^{good} will be deficiency clen^3 subsources of X^{med} and Y. We will ensure that in the final subsources, for every element $w \in \mathcal{L}_{v^{\mathsf{b}}}$, $\mathsf{Challenge}(X^{\mathsf{good}}_w, Y^{\mathsf{good}})$ is clen-responded by the response $\mathsf{Response}(X^{\mathsf{good}}_{\mathsf{par}(w)}, Y^{\mathsf{good}})$ with probability 1.

First we will show that we can move to a subsource where the relevant challenges are fixed.

Claim 6.11. *There is a subsource $Y' \subset Y$ of deficiency at most $\mathsf{t} \cdot \mathsf{len} \cdot \log n$ s.t. every challenge $\mathsf{Challenge}(X^{\mathsf{med}}_w, Y')$ for $w \in \mathcal{L}_{v^{\mathsf{b}}}$ is fixed to a constant string in the subsources X^{med}, Y'.*

Proof. By the **{Fixed left family}** property after Step 1, we have that for every $w \in \mathcal{L}_{v^{\mathsf{b}}}$, X^{med}_w is fixed. Note that $\mathsf{Challenge}(X^{\mathsf{med}}_w, Y)$ is a function only of X^{med}_w and Y. Thus, for every $w \in \mathcal{L}_{v^{\mathsf{b}}}$, $\mathsf{Challenge}(X^{\mathsf{med}}_w, Y)$ is a function only of Y.

There are at most $|\mathcal{L}_{v^{\mathsf{b}}}| \leq \mathsf{t} \log n$ challenges to consider, each of length len bits. Thus by Lemma 3.13, we can ensure that there is a deficiency $\mathsf{t} \cdot \mathsf{len} \cdot \log n$ subsource $Y' \subset Y$ in which all the challenges are also fixed. $\square$

Next we will prove that there are even smaller subsources in which each of these challenges is responded with probability 1.

Claim 6.12. *There are subsources $X^{\mathsf{good}} \subset X^{\mathsf{med}}$ and $Y^{\mathsf{good}} \subset Y'$ of deficiency at most $O(\mathsf{t} \cdot \mathsf{len} \cdot \log n)$ in which every challenge $\mathsf{Challenge}(X^{\mathsf{good}}_w, Y^{\mathsf{good}}), w \in \mathcal{L}_{v^{\mathsf{b}}}$ is clen-responded with probability 1 by the response $\mathsf{Response}(X^{\mathsf{good}}_{\mathsf{par}(w)}, Y^{\mathsf{good}})$.*

Proof. Let $\mathcal{L}_{v^{\mathsf{b}}} = \{w_1, w_2, \ldots, w_d\}$. We will prove the stronger statement that for every i with $1 \le i \le d$, there are subsources $X'' \subset X^{\mathsf{med}}, Y'' \subset Y'$ of deficiency at most $2\mathsf{len} i$ in which each $\mathsf{Challenge}(X''_{w_j}, Y'')$ is clen-responded by $\mathsf{Response}(X''_{\mathsf{par}(w_j)}, Y'')$ for $1 \le j \le i$. We prove this by induction on i.

For the base case of $i = 1$, note that $\mathsf{Challenge}(X^{\mathsf{med}}_{w_1}, Y')$ is fixed to a constant in the source X^{med}. Since $H_\infty(X^{\mathsf{med}}_{\mathsf{par}(w_1)}) \ge H_\infty(X^{\mathsf{med}}_{v^{\mathsf{b}}}) \ge k^{0.9}$ and $H_\infty(Y') \ge k - \mathsf{t} \cdot \mathsf{len} \cdot \log n \ge k^{0.9}$, we get that $X^{\mathsf{med}}_{\mathsf{par}(w)}, Y'$ are sources that have enough entropy for our somewhere extractor SE to succeed. By the **{Hitting matrices}** property of Corollary 6.3, we can then ensure that there are deficiency $2\mathsf{len}$ subsources $X'' \subset X^{\mathsf{med}}, Y'' \subset Y'$ in which $\mathsf{Challenge}(X''_{w_1}, Y'')$ is clen-responded by the $\mathsf{Response}(X''_{\mathsf{par}(w_1), Y''}$ with probablity 1.

For $i > 1$, we use the inductive hypothesis to find subsources $\hat{X} \subset X^{\mathsf{med}}, \hat{Y} \subset Y'$ of deficiency at most $2\mathsf{len}(i-1)$ on which all the previous challenges are clen-responded. Then, since $H_\infty(\hat{X}_{\mathsf{par}(w_i)}) \ge H_\infty(X^{\mathsf{med}}_{v^{\mathsf{b}}}) - 2\mathsf{len}(i-1) \ge k^{0.9}$ and $H_\infty(\hat{Y}) \ge k - \mathsf{t} \cdot \mathsf{len} \cdot \log n - 2\mathsf{len}(i-1) \ge k^{0.9}$, we get that $\hat{X}_{\mathsf{par}(w)}, \hat{Y}$ are sources that have enough entropy for our somewhere extractor SE to succeed. Thus we can find deficiency $2\mathsf{len} \cdot i$ subsources $X'' \subset X^{\mathsf{med}}, Y'' \subset Y'$ in which even $\mathsf{Challenge}(X''_{w_i}, Y'')$ is clen-responded by $\mathsf{Response}(X''_{\mathsf{par}(w_i)}, Y'')$. $\square$

Together the claims give that $X^{\mathsf{good}} \subset X^{\mathsf{med}}, Y^{\mathsf{good}} \subset Y$ are subsources in which all the challenges of the left family are responded with probability 1 and are of deficiency at most $O(\mathsf{t} \cdot \mathsf{len} \cdot \log n) < \mathsf{clen}^{2.1}$ by our choice of parameters.

Since we only went down to a $\mathsf{clen}^{2.1}$ deficiency subsource of X^{med} in all of these steps, by Corollary 3.19, we still retain the block-source structure of $X^{\mathsf{med}}_{v^{\mathsf{b}}}$. In particular, the corollary implies that $X^{\mathsf{good}}_{v^{\mathsf{b}}}$ is $2^{-19\mathsf{clen}^3}$ close to being a $(\mathsf{C}, k^{0.9} - 20\mathsf{clen}^3 \ge k^{0.8})$-somewhere block-source.

Similarly $H_\infty(X^{\mathsf{good}}_{v^{\mathsf{med}}}) \ge H_\infty(X^{\mathsf{med}}_{v^{\mathsf{med}}}) - \mathsf{clen}^3 \ge k^{0.9} - \mathsf{clen}^3 \ge k^{0.8}$ and conditioned on any fixing of $X^{\mathsf{good}}_{v^{\mathsf{med}}}$, $H_\infty(X^{\mathsf{good}}_{\mathsf{par}(v^{\mathsf{med}})}) \ge k^{0.9}$, since $X^{\mathsf{med}}_{\mathsf{par}(v^{\mathsf{med}})}$ was shown to be a block-source with min-entropy $k^{0.9}$.

Result of Step 2: At this point we have X^{good} and Y^{good}, which are deficiency $k/4 + \mathsf{clen}^3$ subsources of the sources X and Y satisfying:

$X^{\mathsf{good}}_{v^{\mathsf{med}}} \circ X^{\mathsf{good}}$ **is a block-source:** $H_\infty(X^{\mathsf{good}}_{v^{\mathsf{med}}}) \ge k^{0.8}$ and $X^{\mathsf{good}}_{\mathsf{par}(v^{\mathsf{med}})}$ has entropy greater than $k^{0.9}$ even conditioned on any fixing of $X^{\mathsf{good}}_{v^{\mathsf{med}}}$.

$X^{\mathsf{good}}_{v^{\mathsf{b}}}$ **has a block-source:** The source $X^{\mathsf{good}}_{v^{\mathsf{b}}}$ is $2^{-19\mathsf{clen}^3}$ close to being a $(\mathsf{C}, k^{0.8})$-somewhere block-source.

Low blocks are correctly identified: For every $w \in \mathcal{L}_{v^{\mathsf{b}}}$ $\mathsf{Challenge}(X^{\mathsf{good}}_w, Y^{\mathsf{good}})$ is clen-responded with probability 1 by $\mathsf{Response}(X^{\mathsf{good}}_{\mathsf{par}(w)}, Y^{\mathsf{good}})$.

6.3.3 Step 3: Ensuring that challenges along the path are somewhere random

We argue that in $X^{\mathsf{good}}, Y^{\mathsf{good}}$, for every $w \in \mathcal{P}_{v^{\mathsf{b}}}$, $\mathsf{Challenge}(X^{\mathsf{good}}_w, Y^{\mathsf{good}})$ is $2^{\log^2 n}(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$-close to having min-entropy clen. In fact something even stronger is true:

Lemma 6.13 (The challenges along the good path are somewhere random)**.** *Let $X' \subset X^{\mathsf{good}}, Y' \subset Y^{\mathsf{good}}$ be any deficiency $20\mathsf{len}$ subsources. Then in these subsources, if $w \in \mathcal{P}_{v^{\mathsf{b}}}$ is an ancestor of v^{b}, $\mathsf{Challenge}(X'_w, Y')$ is $2^{\log^2 n}(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$-close to being somewhere random.*

Proof. We will prove the lemma by induction on the vertices in $\mathcal{P}_{v^{\mathsf{b}}}$, starting from v^{b} and moving up the path.

Let h be the depth of v^{b} in the tree (note that $h = O(\log n)$). Let ℓ be the number of matrices in the output of $\mathsf{Response}$ (note that $\ell = \mathrm{poly}(n)$ by Corollary 6.3). For $w \in \mathcal{P}_{v^{\mathsf{b}}}$ at a distance of i from v^{b}, we will prove that as long as $X' \subset X^{\mathsf{good}}, Y' \subset Y^{\mathsf{good}}$ are of deficiency at most $(h-i-1)20\mathsf{len}$, $\mathsf{Challenge}(X'_w, Y')$ is $(2\ell)^i(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$-close to being somewhere random.

For the base case note that by Corollary 3.19, $X'_{v^{\mathsf{b}}}$ is $2^{-19\mathsf{clen}^3} + 2^{-20\mathsf{clen}^3} < 2^{-18\mathsf{clen}^3}$-close to being a $(\mathsf{C}, k^{0.8} - (h-1)20\mathsf{len} - 20\mathsf{clen}^3 > \sqrt{k})$ somewhere block-source and Y' is an independent source with min-entropy $k - (k/4 + \mathsf{clen}^3 + (h-1)20\mathsf{len}) > \sqrt{k}$. Thus, in the subsources X', Y', $\mathsf{Challenge}(X'_{v^{\mathsf{b}}}, Y')$ is $2^{-18\mathsf{clen}^3} + 2^{-k^{c_1}} < (2^{-\mathsf{clen}} + 2^{-k^{c_1}})$-close to being somewhere random by Corollary 6.2.

Now let w be an ancestor of v^{b} and let w' be its child on the path to v^{b}. We want to show that the challenge has entropy even on deficiency $(h-i-1)20\mathsf{len}$ subsources $X' \subset X^{\mathsf{good}}, Y' \subset Y^{\mathsf{good}}$.

We will show that with high probability $\mathsf{Challenge}(X'_w, Y')$ contains $\mathsf{Challenge}(X'_{w'}, Y')$ as a substring. By the induction hypothesis we will then get that $\mathsf{Challenge}(X'_w, Y')$ must be statistically close to being somewhere random also. By our construction, to ensure that this happens we merely need to ensure that $\mathsf{Challenge}(X'_{w'}, Y')$ is clen unresponded by $\mathsf{Response}(X'_w, Y')$. We will argue this using the union bound. Fix an index j and consider the j'th response string $\mathsf{Response}(X'_w, Y')_j$.

By the **{Fixed matrices on low deficiency subsources}** property of Corollary 6.3, we get that X', Y' is $2^{-10\mathsf{len}}$ close to a convex combination of independent sources $\hat{X}, \hat{Y}$, where each element of the convex combination is of deficiency at most $20\mathsf{len}$ and the j'th response string $\mathsf{Response}(\hat{X}_w, \hat{Y})_j$ is fixed to a constant on these subsources. Each element of this convex combination then has a deficiency of at most $(h-i-1)20\mathsf{len} + 20\mathsf{len} = (h-(i-1)-1)20\mathsf{len}$ from $X^{\mathsf{good}}, Y^{\mathsf{good}}$.

By the induction hypothesis, we get that $\mathsf{Challenge}(\hat{X}_{w'}, \hat{Y})$ is $(2\ell)^{i-1}(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$-close to being somewhere random. Thus, the probability that $\mathsf{Challenge}(X'_{w'}, Y')$ i responded by $\mathsf{Response}(X'_w, Y')$ is at most $2^{-\mathsf{clen}} + (2\ell)^{i-1}(2^{-k^{c_1}} + 2^{-\mathsf{clen}}) < 2 \cdot (2\ell)^{i-1}(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$. Thus by the union bound over the ℓ response strings, we get that the probability that the challenge is responded is at most $(2\ell)^i(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$.

Note that the length of the path to v^{b} from the root is $o(\log(n))$, so we will need to repeat the induction only $\log(n)$ times. We get that the challenge is $(2\ell)^h(2^{-k^{c_1}} + 2^{-\mathsf{clen}}) < 2^{\log^2 n}(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$-close to being somewhere random. ☐

Result of Step 3: At this point we have X^{good} and Y^{good}, which are deficiency $k/4 + \mathsf{clen}^3$ subsources of the sources X and Y satisfying:

Challenges along the path are somewhere random, even on subsources If $X' \subset X^{\mathsf{good}}, Y' \subset Y^{\mathsf{good}}$ are deficiency $20\mathsf{clen}$ subsources, $\mathsf{Challenge}(X'_w, Y')$ is $2^{\log^2 n}(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$ close to being somewhere random in X', Y', for every vertex $w \in \mathcal{P}_{v^{\mathsf{med}}}$.

6.3.4 Step 4: Ensuring that Disp outputs both 0 and 1

We will ensure that our disperser outputs both 1 and 0 with significant probability. There are two remaining steps:

- We will ensure that in our good subsources $X^{\mathsf{good}}, Y^{\mathsf{good}}$, with high probability (say $1-\gamma$) $\mathsf{val}(X^{\mathsf{good}}_{[1,n]}, Y^{\mathsf{good}}) = \mathsf{val}(X^{\mathsf{good}}_{v^{\mathsf{med}}}, Y^{\mathsf{good}})$.
- We will ensure that in our good subsources $X^{\mathsf{good}}, Y^{\mathsf{good}}$, $\mathsf{val}(X^{\mathsf{good}}_{v^{\mathsf{med}}}, Y^{\mathsf{good}})$ is both 0 and 1 with significant probability (say $\gamma^{1/10}$).

By the union bound these two facts imply that the disperser outputs both 0 and 1 with positive probability.

Lemma 6.14. *For every vertex v on the path from v^{med} to the root and for any $1 \le q \le \mathsf{clen}$,*

$$\Pr[\mathsf{Challenge}(X_v^{\mathsf{good}}, Y^{\mathsf{good}}) \text{ is } q\text{-responded by } \mathsf{Response}(X_{\mathsf{par}(v)}^{\mathsf{good}}, Y^{\mathsf{good}})] \le 2^{-q} + 2^{\log^2 n}(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$$

Proof. By the **{Fixed matrices on low deficiency subsources}** property of Corollary 6.3, we get that $X^{\mathsf{good}}, Y^{\mathsf{good}}$ is $2^{-10\mathsf{len}}$-close to a convex combination of independent sources, where each element X', Y' of the convex combination is of deficiency at most $20\mathsf{len}$ and the j'th response string $\mathsf{Response}(X'_{\mathsf{par}(v)}, Y')_j$ is fixed to a constant on these subsources. Thus by Lemma 6.13,

$$\Pr[\mathsf{Challenge}(X'_v, Y') \text{ is } q\text{-responded by } \mathsf{Response}(X'_{\mathsf{par}(v)}, Y')] < 2^{-q} + 2^{\log^2 n}(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$$

□

Lemma 6.15 ($\mathsf{val}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}})$ propagates to the root)**.** *Let h be the depth of v^{med} in the tree. Then*

$$\Pr_{X^{\mathsf{good}}, Y^{\mathsf{good}}}[\mathsf{val}(x_{v^{\mathsf{med}}}, y) \neq \mathsf{val}(x_{[1,n]}, y)] < 2^{-\mathsf{clen}_{h,0}}$$

Proof. We will show that for every $w \in \mathcal{P}_{v^{\mathsf{med}}}, w \neq [1, n]$, $\Pr[\mathsf{val}(X_w^{\mathsf{good}}, Y^{\mathsf{good}}) \neq \mathsf{val}(X_{\mathsf{par}(w)}^{\mathsf{good}}, Y^{\mathsf{good}})] < 2^{-\mathsf{clen}_{h,0}}/\log^2 n$. Then we will apply a union bound over all the edges in the path from the root to v^{med} to get the bound for the lemma.

Let h' be the depth of w in the tree. Now note that by our construction

$$\begin{aligned}
&\Pr[\mathsf{val}(X_w^{\mathsf{good}}, Y^{\mathsf{good}}) \neq \mathsf{val}(X_{\mathsf{par}(w)}^{\mathsf{good}}, Y^{\mathsf{good}})] \\
&< \Pr[\mathsf{Challenge}(X_w^{\mathsf{good}}, Y^{\mathsf{good}}) \text{ is } \mathsf{clen}_{h',2}\text{-responded by } \mathsf{Response}(X_{\mathsf{par}(w)}^{\mathsf{good}}, Y^{\mathsf{good}})] \\
&\le 2^{-\mathsf{clen}_{h',2}} + 2^{\log^2 n}(2^{-k^{c_1}} + 2^{-\mathsf{clen}})
\end{aligned}$$

Where the last inequality is by Lemma 6.14. Using the union bound over all poly(n) response strings, we then get that the probability that the challenge is responded is at most $\mathrm{poly}(n)(2^{-\mathsf{clen}_{h',2}} + 2^{\log^2 n}(2^{-k^{c_1}} + 2^{-\mathsf{clen}}) + 2^{-10\mathsf{len}}) < (1/\log^2 n)2^{-\mathsf{clen}_{h,0}}$ by our choice of parameters. Aplying a union bound over the path from the root of the tree to v^{med}, we get the bound claimed by the lemma. □

Finally we argue that the probability that $\mathsf{val}(x_{v^{\mathsf{med}}}, y)$ is 0 or 1 is significantly higher than $2^{-\mathsf{clen}_{h,0}}$. We do this by showing that for any q, the probability that $\mathsf{Challenge}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}})$ is q-responded by $\mathsf{Response}(X_{\mathsf{par}(v^{\mathsf{med}})}^{\mathsf{good}}, Y^{\mathsf{good}})$ can be bounded from above and below:

Lemma 6.16. *Let $p = \Pr[\mathsf{Challenge}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}})$ is q-responded by $\mathsf{Response}(X_{\mathsf{par}(v^{\mathsf{med}})}^{\mathsf{good}}, Y^{\mathsf{good}})]$. Then,*

$$2^{-q \cdot \mathsf{nrows}} - 2^{-10\mathsf{len}} - 2^{-20\mathsf{len}} \le p \le 2^{-q} + 2^{\log^2 n}(2^{-k^{c_1}} + 2^{-\mathsf{clen}})$$

Proof. In Step 2 of the analysis we showed that $X_{v^{\mathsf{med}}}^{\mathsf{good}} \circ X_{\mathsf{par}(v^{\mathsf{med}})}^{\mathsf{good}}$ is block-source with block entropy $k^{0.9}$. Thus X^{good} is a convex combination of sources where for every element of the combination $\hat{X}$,

- $\hat{X}_{v^{\mathsf{med}}}$ is fixed
- $\hat{X}_{\mathsf{par}(v^{\mathsf{med}})}$ has min-entropy $k^{0.8}$

For every such subsource $\hat{X}$, $\mathsf{Challenge}(\hat{X}_v^{\mathsf{med}}, Y^{\mathsf{good}})$ is a function only of Y^{good}. Thus by Lemma 3.13, for every such subsource $\hat{X}$, Y^{good} is $2^{-20\mathsf{len}}$ close to a convex combination of sources where for each element of the combination $\hat{Y}$ is of deficiency at most $21\mathsf{len}$ and $\mathsf{Challenge}(\hat{X}_v^{\mathsf{med}}, \hat{Y})$ is fixed to a constant. Thus overall we get a convex combination of sources where for each element of the convex combination:

- In $\hat{X}, \hat{Y}$, $\mathsf{Challenge}(\hat{X}_v^{\mathsf{med}}, \hat{Y})$ is fixed.
- $\hat{X}_{\mathsf{par}(v^{\mathsf{med}})}, \hat{Y}$ are independent sources with min-entropy $k^{0.8}$ each.

By Corollary 6.3 we get that $\mathsf{Response}(\hat{X}_{\mathsf{par}(v^{\mathsf{med}})}, \hat{Y})$ is $2^{-10\mathsf{len}}$-close to being somewhere random, implying that the challenge is q-responded with probability at least $2^{-q\cdot\mathsf{nrows}} - 2^{-10\mathsf{len}}$ in these subsources. Thus we get that $\Pr \mathsf{Challenge}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}})$ is q-responded by $\mathsf{Response}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}})] \geq 2^{-q\cdot\mathsf{nrows}} - 2^{-10\mathsf{len}} - 2^{-20\mathsf{len}}$.

The upper bound follows from Lemma 6.14. $\square$

This lemma then implies that $\mathsf{val}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}})$ takes on both values with significant probability:

Lemma 6.17 ($\mathsf{val}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}})$ is both 0 and 1 with significant probability)**.**

$$\Pr[\mathsf{val}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}}) = 1] > (0.5)2^{-\mathsf{len}_{h,1}}$$

$$\Pr[\mathsf{val}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}}) = 0] > (0.5)2^{-\mathsf{len}_{h,2}}$$

Proof. Note that

$$\begin{aligned}
&\Pr[\mathsf{val}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}}) = 1] \\
&\geq \Pr[\mathsf{Challenge}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}}) \text{ is } \mathsf{clen}_{h,1}\text{-responded by } \mathsf{Response}(X_{\mathsf{par}(v^{\mathsf{med}})}^{\mathsf{good}}, Y^{\mathsf{good}})] \\
&- \Pr[\mathsf{Challenge}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}}) \text{ is } \mathsf{clen}_{h,0}\text{-responded by } \mathsf{Response}(X_{\mathsf{par}(v^{\mathsf{med}})}^{\mathsf{good}}, Y^{\mathsf{good}})] \\
&\geq 2^{-\mathsf{clen}_{h,1}\cdot\mathsf{nrows}} - 2^{-10\mathsf{len}} - 2^{-20\mathsf{len}} \\
&- 2^{-\mathsf{clen}_{h,0}} + 2^{\log^2 n}(2^{-k^{c_1}} + 2^{-\mathsf{clen}}) \\
&\geq 2^{-\mathsf{len}_{h,1}} - 2^{-10\mathsf{len}} - 2^{-20\mathsf{len}} - 2\cdot 2^{-\mathsf{clen}_{h,0}} \\
&\geq (0.5)2^{-\mathsf{len}_{h,1}}
\end{aligned}$$

Similarly,

$$\begin{aligned}
&\Pr[\mathsf{val}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}}) = 0] \\
&\geq \Pr[\mathsf{Challenge}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}}) \text{ is } \mathsf{clen}_{h,2}\text{-responded by } \mathsf{Response}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}})] \\
&- \Pr[\mathsf{Challenge}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}}) \text{ is } \mathsf{clen}_{h,1}\text{-responded by } \mathsf{Response}(X_{v^{\mathsf{med}}}^{\mathsf{good}}, Y^{\mathsf{good}})] \\
&\geq 2^{-\mathsf{len}_{h,2}} - 2^{-10\mathsf{len}} - 2^{-20\mathsf{len}} - 2\cdot 2^{-\mathsf{clen}_{h,1}} \\
&> (0.5)2^{-\mathsf{len}_{h,2}}
\end{aligned}$$

$\square$

This concludes the proof that $\mathsf{Disp}(X, Y)$ outputs both zero and one proving Theorem 1.10.

7 Proof of Theorem 4.1

In this section we prove Theorem 4.1 (that gives an extractor for one block-wise source and one general source). Our techniques rely on those of Rao [Rao06]. In particular, we will obtain our extractor by reducing the problem to the one of constructing an extractor for two independent somewhere random sources, a problem that was solved in [Rao06].

We first discuss the new ideas that come into obtaining the improvement in the error parameter (which can be also be applied directly to Rao's [Rao06] extractor). We then give the full construction for the new extractor.

7.1 Achieving Small Error

The lower error is achieved by a careful analysis of our construction. A somewhat similar observation was made by Chung and Vadhan [CV], who noted that the construction of Rao can more directly be shown to have low error.

In our construction, we will actually prove the following theorem, which gives an extractor for a block-source and an independent somewhere random source.

Theorem 7.1 (Somewhere random + Block-source Extractor)**.** *There exist constants $\alpha, \beta, \gamma < 1$ and a polynomial time computable function $\mathsf{SR}+\mathsf{BExt} : \{0,1\}^{\mathsf{C}n} \times \{0,1\}^{tk} \to \{0,1\}^m$ such that for every n, t, k, with $k > \log^{10} t, k > \log^{10} n$ with $\mathsf{C} = O(\frac{\log t}{\log k})$ s.t. , if $X = X^1 \circ \cdots \circ X^{\mathsf{C}}$ is a $(k, \dots, k)$ block-source and Y is an independent $(t \times k)$ $(k - k^{\beta})$-SR-source,*

$$|X \circ \mathsf{SR}+\mathsf{BExt}(X, Y) - X \circ U_m| < \epsilon$$

$$|Y \circ \mathsf{SR}+\mathsf{BExt}(X, Y) - Y \circ U_m| < \epsilon$$

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

Note that we can get an extractor from a block-source and a general independent source from Theorem 7.1 by using the fact that a general source can be tranformed into a somewhere random source (Proposition 3.22). However, using this transformation spoils the error, since the transformation has only polynomially small error. In order to bypass this difficulty, we use a more careful analysis. We first use Theorem 7.1 to prove the following theorem which is weaker than Theorem 4.1. We will then obtain Theorem 4.1.

Theorem 7.2 (Block + Arbitrary Source Extractor)**.** *There exist absolute constants $c_1, c_2, c_3 > 0$ and a polynomial time computable function $\mathsf{BExt} : \{0,1\}^{\mathsf{C}n} \times \{0,1\}^{n'} \to \{0,1\}^m$ such that for every n, n', k, with $k > \log^{10}(n + n')$ with $\mathsf{C} = c_1 \frac{\log n}{\log k}$, such that if $X = X^1 \circ \cdots \circ X^{\mathsf{C}}$ is a k block-source and Y is an independent (n', k)-source, there is a deficiency 2 subsource $Y' \subseteq Y$ s.t.*

$$|X \circ \mathsf{BExt}(X, Y') - X \circ U_m| < \epsilon$$

$$|Y' \circ \mathsf{BExt}(X, Y') - Y' \circ U_m| < \epsilon$$

where U_m is independent of X and Y, and for $m = c_2 k$ and $\epsilon = 2^{-k^{c_3}}$.

Proof. The idea is to reduce to the case of Theorem 7.1. We convert the general source Y into an SR-source. To do this we will use a strong seeded extractor and the Proposition 3.22. If we use a strong seeded extractor that requires only $O(\log n)$ bits of seed, the SR-source that we get will have only $\mathrm{poly}(n)$ rows. This adds a polynomial amount of error. By Lemma 3.15, we can go to a deficiency 2 subsource $Y' \subseteq Y$ which has high entropy in some row. This is good enough to use our extractor from Theorem 7.1 and get the better error. □

Proof of Theorem 4.1. We prove the theorem by showing that any extractor that satisfies the conclusions of Theorem 7.2 (i.e. low strong error on a subsource), must satisfies the seemingly stronger conclusions of Theorem 4.1.

Let BExt be the extractor from Theorem 7.2, set up to extract from a $k/2$ block-source and a $k/2 - 2$ general source. Then we claim that when this extractor is run on a k block-source and a k general source, it must succeed with much smaller error.

Given the source X let $B_X \subset \{0,1\}^{n'}$ be defined as $B_X = \{y : |\mathsf{BExt}(X, y) - U_m| \geq \epsilon\}$. Then,

Claim 7.3. $|B_X| < 2^{k/2}$

Proof. The argument for this is by contradiction. Suppose $|B_X| \geq 2^{k/2}$. Then define Z to be the source which picks a uniformly random element of B_X. By the definition of B_X, this implies that $|Z' \circ \mathsf{BExt}(X, Z') - Z' \circ U_m| \geq \epsilon$ for *any* subsource $Z' \subset Z$. This contradicts Theorem 7.2. □

Thus $\Pr[Y \in B_X] < 2^{k/2-k} = 2^{-k/2}$.

This implies that $|\mathsf{BExt}(X, Y) - U_m| < \epsilon + 2^{-k/2}$, where ϵ is the ϵ from Theorem 7.2. □

Remark 7.4. In fact the above proof actually implies the extractor from Theorem 4.1 is *strong* with respect to Y, i.e. $|Y \circ \mathsf{BExt}(X, Y) - Y \circ U_m| < \epsilon + 2^{-k/2}$.

7.2 Extractor for general source and an SR-source with few rows

Here we will construct the extractor for Theorem 7.1. The main step in our construction is the construction of an extractor for a general source and an independent SR-source which has few rows. Once we have such an extractor, it will be relatively easy to obtain our final extractor by iterated condensing of SR-sources.

First, we prove the following theorem:

Theorem 7.5. *There are constants $\alpha, \beta < 1$ and a polynomial time computable function* $\mathsf{BasicExt}$: $\{0,1\}^n \times \{0,1\}^{k^{\gamma+1}} \to \{0,1\}^m$ *such that for every $n, k(n)$ with $k > \log^{10} n$, and constant $0 < \gamma < 1/2$, if X is an (n, k) source and Y is a $(k^\gamma \times k)$ $(k - k^\beta)$-SR-source,*

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

and

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

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

Proof. We are trying to build an extractor that can extract from one $(k^\gamma \times k)$ k^β-SR-source Y and an independent (n, k) source X. We will reduce this to the case of two independent aligned SR-sources with few rows, for which we can use Theorem 3.27.

The plan is to use the structure in the SR-source Y to impose structure on the source X. We will first use Y and X to get a list of candidate seeds, such that one seed in the list is close to uniformly random and independent of both X and Y. Once we have this list, we can readily reduce the problem to that of extracting from independent aligned SR-sources with few rows.

In the following discussion, the term *slice* refers to a subset of the bits coming from an SR-source that takes a few bits of the SR-source from every row (Definition 3.8). We also remind the reader of the following notation: if $f : \{0,1\}^r \times \{0,1\}^r \to \{0,1\}^m$ is a function and a, b are samples from $(t \times r)$ somewhere sources, $f(\vec{a}, \vec{b})$ refers to the $(t \times m)$ matrix whose ith row is $f(a_i, b_i)$. Similarly, if c is an element of $\{0,1\}^r$ and b is a sample from a $(t \times r)$ somewhere source, $f(c, \vec{b})$ refers to the $(t \times m)$ matrix whose ith row is $f(c, b_i)$.

We first write down the algorithm for our extractor. Then we shall describe the construction in words and give more intuition.

Algorithm 7.6.
$\mathsf{BasicExt}(x, y)$

Input: x, a sample from an (n, k) source and y a sample from a $(k^\gamma \times k)$ k^β-somewhere random source.

Output: z

Let $w, w', w'', l, d, \beta_1$ be parameters that we will pick later. These will satisfy $w'' > w > k^\gamma$ and $w - k^\gamma > w'$.

Let $\mathsf{Raz}_1 : \{0,1\}^n \times \{0,1\}^w \to \{0,1\}^{w'}$ be the extractor from Theorem 3.26 setup to extract w' bits from an (n, k) source, using a $(w, 0.9w)$ source as seed.

Let $\mathsf{Raz}_2 : \{0,1\}^{w'} \times \{0,1\}^{w''} \to \{0,1\}^d$ be the extractor from Theorem 3.26, setup to extract d bits from a (w', w') source and an independent $(w'', 0.9w'')$ source.

Let $\mathsf{Ext}_1 : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^{k-k^{\beta_1}}$ and $\mathsf{Ext}_2 : \{0,1\}^{k^{1+\gamma}} \times \{0,1\}^d \to \{0,1\}^{k-2k^{\beta_1}}$ be strong seeded extractors from Theorem 3.24, each set up to extract from min-entropy $k - k^{\beta_1}$ with error $2^{-k^{\Omega(1)}}$.

Let $\mathsf{2SRExt} : \{0,1\}^{k^\gamma(k-2k^{\beta_1})} \times \{0,1\}^{k^\gamma(k-2k^{\beta_1})} \to \{0,1\}^m$ be the extractor from Theorem 3.27, setup to extract from two aligned $(k^\gamma \times k - 2k^{\beta_1})$ SR-sources.

Let Slice be the function defined in Definition 3.8.

1. Set $s = \mathsf{Slice}(y, w)$.
2. Treating s as a list of k^γ seeds, use it to extract from x to get $q = \mathsf{Raz}_1(x, \vec{s})$. The result is a string with k^γ rows, each of length w'.
3. Set $r = \mathsf{Slice}(y, w'')$.
4. Let $h = \mathsf{Raz}_2(\vec{q}, \vec{r})$, i.e. h is a list of k^γ strings, where the ith string is $\mathsf{Raz}_2(q_i, r_i)$.
5. Let $x' = \mathsf{Ext}_1(x, \vec{h}), y' = \mathsf{Ext}_2(y, \vec{h})$.
6. Use $\mathsf{2SRExt}$ to get $z = \mathsf{2SRExt}(x', y')$.

The first target in the above algorithm is to generate a list of candidate seeds (S) from the sources, one of which will be close to uniformly random. To generate the list of seeds that we want, we will first take a small slice of the bits from Y, i.e. we take $\mathsf{Slice}(Y, w)$, where w is a parameter that we will pick later (think of w as k^μ for small μ). We will be able to guarantee that at least one of the rows of $\mathsf{Slice}(Y, w)$ has high entropy. We can then use Raz's extractor Theorem 3.26 with these bits to extract from X. This gives us a $(k^\gamma \times w')$ SR-source Q, where $w' = k^{\theta(1)} \gg w$ is some parameter that we will pick later. The two sources that we have now (Q and Y) are not independent, but note that when we fix the slice of bits (S) that we used, we get two independent sources. Y conditioned on the value of

S could potentially lose entropy in its high entropy row. Still, we can expect this high entropy row to have about $k - k^\beta - wk^\gamma$ bits of entropy, since we fixed only wk^γ bits of Y in S. In the next step we take a wider slice of Y and call it $R = \mathsf{Slice}(Y, w'')$. Note that on fixing S to a typical value, we get that Q, R are two independent aligned somewhere high entropy sources. We then use Raz's extractor again to convert Q, R into a somewhere random source H, by applying the extractor to each pair of rows from Q, R. Since Raz's extractor is strong, we will be able to guarantee that one of the rows in the resulting SR-source is independent of both input sources. Further, we can fix a random variable which determines the value of H, yet does not break the independence between X, Y.

Thus, once we have H, we can use it with a strong seeded extractor to extract from both X and Y to get independent aligned SR-sources of the type that Theorem 3.27 can handle.

We will prove the following lemma:

Lemma 7.7. *For every (n, k) source X and a $(k^\gamma \times k)$ k^β-somewhere random source Y as in Theorem 7.5, we can pick $w, w', w'', l, d, \beta_1$ and a constant β s.t. $(X \circ Y)$ is $2^{-k^{\Omega(1)}}$-close to a convex combination of sources s.t. for any source in the convex combination, $(X' \circ Y')$ in step 5 above:*

1. *X' is independent of Y'*
2. *X' is a $(k^\gamma \times k - k^\beta)$ SR-source*
3. *Y' is a $(k^\gamma \times k - k^\beta)$ SR-source*

Given the lemma, we have reduced the problem to one of extracting from aligned somewhere random sources. Theorem 7.5 then follows by the properties of $\mathsf{2SRExt}$.

Proof of Lemma 7.7. We assume that we have some fixed random variables X, Y that satisfy the hypotheses of the lemma. We will make several claims about the various random variables involved in the construction, setting $w, w', w'', l, d, \beta_1$ along the way to ensure that our lemma is true. In the rest of this proof, a capital letter represents the random variable for the corresponding small letter in the construction above.

Recall that k^β (we are allowed to set $\beta < 1$ to anything we want) is the randomness deficiency of the random row in Y. Note that:

Claim 7.8. *For any $w > 2k^\beta$, S is 2^{-k^β} close to a $(k^\gamma \times w)$ $(w - 2k^\beta)$-SR-source*

Proof. This follows from an application of Lemma 3.14. □

We set $w = k^{\alpha_1}$ for some constant α_1 s.t. $\alpha_1 + \gamma < 1$ and $\alpha_1 > \beta$ and set $w' = w/10$. Note that Theorem 3.26 does give an extractor for a $(w, w - 2k^\beta)$ source and an independent (n, k) source with output length $w/10$.

Now Q is correlated with both X and Y. However, when we fix S, Q becomes independent of Y, i.e.: $(X \circ Q)|S = s$ is independent of $Y|S = s$ for any s. Since Raz_1 is a strong extractor, Q still contains a random row for a typical fixing of S.

Claim 7.9. *There exists some constant $\alpha_2 < 1$ s.t. $\Pr_{s \leftarrow_R S}[Q|S = s$ is $2^{-k^{\alpha_2}}$ close to a $(k^\gamma \times w')$ SR-source$] > 1 - 2^{-k^{\alpha_2}}$.*

Thus with high probability Q is independent upto convex combinations from Y.

Next, set $w'' = k^{\alpha_3}$, where $1 > \alpha_3 > \alpha_1 + \gamma$ is any constant. Now consider the random variable R.

Claim 7.10. *R is 2^{-k^β} close to a $(k^\gamma \times w'')$ $(w'' - 2k^\beta)$-SR-source.*

Proof. This follows from an application of Lemma 3.14. □

Now we assume that R is in fact a $w'' - 2k^\beta$-SR-source (we will add 2^{-k^β} to the final error).

After we fix S, R can lose entropy in its random row, but not much. We can expect it to lose as many bits of entropy as there are in S, which is only $k^{\alpha_1+\gamma}$. Since we picked $w'' = k^{\alpha_3} \gg k^{\alpha_1+\gamma}$, we get that R still contains entropy.

Claim 7.11. $\Pr_{s \leftarrow_R S}[R|S{=}s$ *is a* $(k^\gamma \times w'')$ $(w'' - 2k^{\alpha_3})$*-SR-source* $] > 1 - 2^{-k^{\alpha_3}}$.

Proof. By Fact 3.12, we get that $\Pr_{s \leftarrow_{\mathrm{R}} S}[R|S{=}s$ is a $(k^\gamma \times w'')$ $(w'' - k^{\alpha_1+\beta} - l)$-SR-source $] > 1 - 2^l$. Setting $l = k^{\alpha_3}$ gives the claim. □

Thus, upto a typical fixing of S, (Q, R) are statistically close to two aligned sources, Q a $(k^\gamma \times w')$ SR-source, and R an independent $(k^\gamma \times w'')$ $(0.1w'')$-SR source. If we set $d = w'/10$, we see that our application of Raz_2 above succeeds. In the aligned good row, Raz_2 gets two independent (after fixing S) sources which are statistically close to having extremely high entropy.

The result of applying Raz_2 is the random variable H.

Claim 7.12. *H is $2^{-\Omega(d)}$ close to a $(k^\gamma, \Omega(d))$ SR-source.*

In addition, we argue that the random row of H is independent of both X and Y. Without loss of generality, assume that H^1 is the random row of H. Let $\alpha_4 > 0$ be a constant s.t. $2^{-k^{\alpha_4}}$ is an upperbound on the error of $\mathsf{Ext}_1, \mathsf{Ext}_2$. Then for a typical fixing of Q, R, we get that X, Y are independent sources, and the random row of H (which is determined by (Q, R)) is a good seed to extract from both sources.

Claim 7.13. *With high probability H contains a good seed to extract from each of the sources:*

$$\Pr_{(q,r) \leftarrow_R (Q,R)}[|\mathsf{Ext}_2((Y|R{=}r), h^1(q,r)) - U_m| \geq 2^{-k^{\alpha_4}}]] < 2^{-k^{\alpha_4}}$$

and

$$\Pr_{(q,r) \leftarrow_R (Q,R)}[|\mathsf{Ext}_1((X|S{=}s(r), Q{=}q), h^1(q,r)) - U_m| \geq 2^{-k^{\alpha_4}}] < 2^{-k^{\alpha_4}}$$

Sketch of proof. There are two ways in which the claim can fail. Either S, Q, R steal a lot of entropy from X, Y, or they produce a bad seed in H to extract from $X|S{=}s, Q{=}q$ or $Y|R{=}r$. Both events happen with small probability.

Specifically,we have that there exist constants β_1, β_2 s.t.

- By Lemma 3.13, $\Pr_{r \leftarrow_{\mathrm{R}} R}[H_\infty(Y|R = r) < k - k^{\beta_1}] < 2^{-k^{\beta_2}}$
- By Lemma 3.13, $\Pr_{(q,r) \leftarrow_{\mathrm{R}} R}[H_\infty(X|R = r, Q = q) < k - k^{\beta_1}] < 2^{-k^{\beta_2}}$
- By our earlier claims, $\Pr_{r \leftarrow_{\mathrm{R}} R}[H|R = r$ is $2^{-k^{\beta_2}}$-close to being somewhere random$]$
- By our earlier claims, $\Pr_{(s,q) \leftarrow_{\mathrm{R}} (S,Q)}[H|S = s, Q = q$ is $2^{-k^{\beta_2}}$-close to being somewhere random$]$
- By the properties of the strong seeded extractor Ext_1, for any s, q such that $H_\infty(X|S = s, Q = q) \geq k - k^{\beta_1}$ and $H|S = s, Q = q$ is $2^{-k^{\beta_2}}$-close to being somewhere random,

$$\Pr_{h \leftarrow_{\mathrm{R}} H|Q=q,S=s}[|\mathsf{Ext}_1((X|S = s, Q = q), (H|S = s, Q = q)) - U_m| \geq 2^{-k^{\beta_2}}] < 2 \cdot 2^{-k^{\beta_2}}$$

- By the properties of the strong seeded extractor Ext_2, for any r such that $H_\infty(Y|R=r) \geq k - k^{\beta_1}$ and $H|R=r$ is $2^{-k^{\beta_2}}$-close to being somewhere random,

$$\Pr_{h \leftarrow_{\mathrm{R}} H|R=r}[|\mathsf{Ext}_2((Y|R=r),(H|R=r)) - U_m| \geq 2^{-k^{\beta_2}}] < 2 \cdot 2^{-k^{\beta_2}}$$

Thus we can use the union bound to get our final estimate. □

□

This concludes the proof of Theorem 7.5. □

Proof of Theorem 7.1. As in [Rao06], the theorem is obtained by repeated condensing of SR-sources. In each condensing step, we will consume one block of X to reduce the number of rows of the SR-source by a factor of $k^{\Omega(1)}$. Thus after $O(\log t/\log k)$ steps, we will have reduced the number of rows to just 1, at which point extraction becomes trivial.

Algorithm 7.14.
$\mathsf{Cond}(x,y)$ Set $\gamma \ll 1/2$ to some constant value. Let β be the constant guaranteed by Theorem 7.1.

For these γ, β, let $\mathsf{BasicExt}$ be the function promised by Theorem 7.5. Let m, ϵ be the output length and error of $\mathsf{BasicExt}$ respectively.

Input: $x = x^1 \circ x^2 \circ \cdots \circ x^{\mathsf{C}}$, a sample from a block-source and y a sample from a $(t \times k)$ SR-source.

Output: $z = x^2 \circ x^3 \circ \cdots \circ x^{\mathsf{C}}$ and y' a $((t/k^\gamma) \times m)$ sample that we will claim comes from a SR-source.

1. Partition the t rows of y equally into t/k^γ parts, each containing k^γ rows. Let $y^{(j)}$ denote the j'th such part.
2. For all $1 \leq j \leq t/k^\gamma$, let $y'_j = \mathsf{BasicExt}(x^1, y^{(j)})$.
3. Let y' be the string with rows $y'_1, y'_2, \ldots, y'_{t/k^\gamma}$.

Given $X = X^1 \circ \cdots \circ X^{\mathsf{C}}$ and Y, the above algorithm uses X^1 to condense Y. Even though this introduces dependencies between X and Y, once we fix X^1, the two output distributions are once again independent. Formally we will argue that after applying the condenser, the output random variables Z and Y' above are statistically close to a convex combination of independent sources, where Z is a block-source with one less block than X, and Y' is an SR-source with much fewer rows than Y.

Lemma 7.15. *Let X, Y be as above. Let ϵ be the error of* $\mathsf{BasicExt}$. *Then $(Z = X^2 \circ \ldots \circ X^{\mathsf{C}}, Y')$ is $2\sqrt{\epsilon}$-close to a convex combination of sources where each source in the combination has*

1. *Z is a $(k, \ldots, k)$ block-source*
2. *Y' is a $(t/k^\gamma, m)$ SR-source*
3. *Z is independent of Y'*

Proof. Let $h \in [t/k^\gamma]$ be such that $Y^{(h)}$ contains the random row. Consider the random variable X^1. We will call x^1 *good* if $|\mathsf{BasicExt}(Y^{(h)}, x^1) - U_m| < \sqrt{\epsilon}$, where m, ϵ are the output length and error of $\mathsf{BasicExt}$ respectively.

Then we make the following easy claims:

Claim 7.16. *For good x^1,*

1. *$Z|X^1 = x^1$ is a $(k, \ldots, k)$ block-source*

2. *$Y'|X^1 = x^1$ is a $\sqrt{\epsilon}$-close to being a $((t/k^\gamma) \times m)$ SR-source*

3. *$Z|X^1 = x^1$ is independent of $Y'|X^1 = x^1$*

Proof. The first and third property are trivial. The second property is immediate from the definition of *good*. □

Claim 7.17. $\Pr[X^1$ *is not* good $] < \sqrt{\epsilon}$

Proof. This is an immediate consequence of Theorem 7.5. □

These two claims clearly imply the lemma. □

Now we use Cond repeatedly until the second source contains just one row. At this point we use the one row with Raz's extractor from Theorem 3.26 with X to get the random bits.

To see that the bits obtained in this way are strong, first note that Raz's extractor is strong in both inputs. Let O be the random variable that denotes the output of our function $\mathsf{BExt}(X, Y)$. Let Q denote the concatenation of all the blocks of X that were consumed in the condensation process. Let U_m denote a random variable that is independent of both X, Y. Then we see that these variables satisfy the hypothesis of Lemma 3.2, i.e. on fixing Q to a good value, Raz's extractor quarantees that the output is independent of both inputs, thus we must have that the output is close to being independent of both inputs. The dominant error term in BExt comes from the first step, when we convert Y to an SR-source. □

8 Open Problems

Better Independent Source Extractors A bottleneck to improving our disperser is the block versus general source extractor of Theorem 2.4. A good next step would be to try to build an extractor for one block-source (with only a constant number of blocks) and one other independent source which works for polylogarithmic entropy, or even an extractor for a constant number of sources that works for sub-polynomial entropy.

Simple Dispersers While our disperser is polynomial time computable, it is not as explicit as one might have hoped. For instance the Ramsey Graph construction of Frankl-Wilson is extremely simple: For a prime p, let the vertices of the graph be all subsets of $[p^3]$ of size $p^2 - 1$. Two vertices S, T are adjacent if and only if $|S \cap T| \equiv -1 \bmod p$. It would be nice to find a good disperser that beats the Frankl-Wilson construction, yet is comparable in simplicity.

9 Acknowledgements

We would like to thank David Zuckerman for useful comments.

References

[Alo98] N. Alon. The Shannon Capacity of a Union. *Combinatorica*, 18, 1998.

[Bar06] B. Barak. A Simple Explicit Construction of an $n^{\tilde{O}(\log n)}$-Ramsey Graph. Technical report, Arxiv, 2006. `http://arxiv.org/abs/math.CO/0601651`.

[BIW04] 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*, pages 384–393, 2004.

[BKS+05] 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*, pages 1–10, 2005.

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

[BKT04] J. Bourgain, N. Katz, and T. Tao. A Sum-Product Estimate in Finite Fields, and Applications. *Geometric and Functional Analysis*, 14:27–57, 2004.

[CRVW02] M. Capalbo, O. Reingold, S. Vadhan, and A. Wigderson. Randomness Conductors and Constant-Degree Lossless Expanders. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pages 659–668, 2002.

[CG88] B. Chor and O. Goldreich. Unbiased Bits from Sources of Weak Randomness and Probabilistic Communication Complexity. *SIAM Journal on Computing*, 17(2):230–261, 1988.

[CV] K.-M. Chung and S. Vadhan. Personal Communication.

[FW81] P. Frankl and R. M. Wilson. Intersection theorems with geometric consequences. *Combinatorica*, 1(4):357–368, 1981.

[Gop06] P. Gopalan. Constructing Ramsey Graphs from Boolean Function Representations. In *Proceedings of the 21th Annual IEEE Conference on Computational Complexity*, 2006.

[Gro00] V. Grolmusz. Low Rank Co-Diagonal Matrices and Ramsey Graphs. *Electr. J. Comb*, 7, 2000.

[Gur03] V. Guruswami. Better Extractors for Better Codes? *Electronic Colloquium on Computational Complexity (ECCC)*, (080), 2003.

[LRVW03] 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*, pages 602–611, 2003.

[MNSW98] P. Miltersen, N. Nisan, S. Safra, and A. Wigderson. On data structures and asymmetric communication complexity. *Journal of Computer and System Sciences*, 57:37–49, 1 1998.

[PR04] P. Pudlak and V. Rodl. Pseudorandom sets and explicit constructions of Ramsey graphs. *Submitted for publication*, 2004.

[Ram28] F. P. Ramsey. On a Problem of Formal Logic. *Proceedings of the London Mathematical Society, Series 2*, 30(4):338–384, 1928.

[Rao06] 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.

[Raz05] R. Raz. Extractors with Weak Random Seeds. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 11–20, 2005.

[RRV02] R. Raz, O. Reingold, and S. Vadhan. Extracting all the Randomness and Reducing the Error in Trevisan's Extractors. *jcss*, 65(1):97–128, 2002.

[RSW00] O. Reingold, R. Shaltiel, and A. Wigderson. Extracting Randomness via Repeated Condensing. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, pages 22–31, 2000.

[SSZ98] M. Saks, A. Srinivasan, and S. Zhou. Explicit OR-Dispersers with Polylog Degree. *Journal of the ACM*, 45:123–154, 1998.

[SV86] M. Santha and U. V. Vazirani. Generating Quasi-Random Sequences from Semi-Random Sources. *Journal of Computer and System Sciences*, 33:75–87, 1986.

[Sha02] R. Shaltiel. Recent Developments in Explicit Constructions of Extractors. *Bulletin of the European Association for Theoretical Computer Science*, 77:67–95, 2002.

[TS02] A. Ta-Shma. Almost Optimal Dispersers. *Combinatorica*, 22(1):123–145, 2002.

[TZ04] A. Ta-Shma and D. Zuckerman. Extractor Codes. *IEEE Transactions on Information Theory*, 50, 2004.

[Tre01] L. Trevisan. Extractors and Pseudorandom Generators. *Journal of the ACM*, pages 860–879, 2001.

[Vaz85] U. Vazirani. Towards a Strong Communication Complexity Theory or Generating Quasi-Random Sequences from Two Communicating Slightly-random Sources (Extended Abstract). In *Proceedings of the 17th Annual ACM Symposium on Theory of Computing*, pages 366–378, 1985.

[WZ99] A. Wigderson and D. Zuckerman. Expanders that Beat the Eigenvalue Bound: Explicit Construction and Applications. *Combinatorica*, 19(1):125–138, 1999.