Electronic Colloquium on Computational Complexity, Report No. 37 (2010)

Simulating Independence: New Constructions of Condensers, Ramsey Graphs, Dispersers, and Extractors*

Boaz Barak† Guy Kindler‡ Ronen Shaltiel§ Benny Sudakov¶ Avi Wigderson‖

March 8, 2010

Abstract

We present new explicit constructions of *deterministic* randomness extractors, dispersers and related objects. We say that a distribution X on binary strings of length n is a δ-source if X assigns probability at most $2^{-\delta n}$ to any string of length n. For every $\delta > 0$ we construct the following poly(n)-time computable functions:

2-source disperser: $D : (\{0,1\}^n)^2 \to \{0,1\}$ such that for any two independent δ-sources X_1, X_2 we have that the support of $D(X_1, X_2)$ is $\{0,1\}$.

Bipartite Ramsey graph: Let $N = 2^n$. A corollary is that the function D is a 2-coloring of the edges of $K_{N,N}$ (the complete bipartite graph over two sets of N vertices) such that any induced subgraph of size N^δ by N^δ is not monochromatic.

3-source extractor: $E : (\{0,1\}^n)^3 \to \{0,1\}$ such that for any three independent δ-sources X_1, X_2, X_3 we have that $E(X_1, X_2, X_3)$ is $o(1)$-close to being an unbiased random bit.

No previous explicit construction was known for either of these for any $\delta < 1/2$, and these results constitute significant progress to long-standing open problems.

A component in these results is a new construction of condensers that may be of independent interest: This is a function $C : \{0,1\}^n \to (\{0,1\}^{n/c})^d$ (where c and d are constants that depend only on δ) such that for every δ-source X one of the output blocks of $C(X)$ is (exponentially close to) a 0.9-source. (This result was obtained independently by Ran Raz).

The constructions are quite involved and use as building blocks other new and known objects. A recurring theme in these constructions is that objects which were designed to work with independent inputs, sometimes perform well enough with correlated, high entropy inputs.

The construction of the disperser is based on a new technique which we call "the challenge-response mechanism" that (in some sense) allows "identifying high entropy regions" in a given pair of sources using only one sample from the two sources.

Categories and Subject Descriptors: G.2.1 [Discrete Mathematics]: Combinatorics

General Terms: Theory.

Keywords: Ramsey Graphs, Explicit Constructions, Extractors, Dispersers, Condensers.

---

*A preliminary version of this paper appeared in STOC 2005.

†Department of Computer Science, Princeton University, boaz@cs.princeton.edu. Supported by NSF grants CNS-0627526, CCF-0426582 and CCF-0832797, US-Israel BSF grant 2004288 and Packard and Sloan fellowships. Most of this work was done when the author was a member in the school of Mathematics at the Institute for Advanced study.

‡Department of Computer Science, Hebrew University, wgkindler@gmail.com. Most of this work was done while the author was member in the School of Mathematics, Institute for Advanced Study.

§Department of Computer Science, University of Haifa, Israel, ronen@cs.haifa.ac.il. Supported by US-Israel BSF grant 2004329 and ISF grant 686/07.

¶Department of Mathematics, University of California at Los Angeles, bsudakov@math.ucla.edu. Supported in part by NSF CAREER award DMS-0812005 and a USA-Israeli BSF grant. Most of this work was done while author was at Princeton University.

‖School of Mathematics, Institute for Advanced Study, Princeton, NJ, avi@ias.edu.

ISSN 1433-8092

Contents

1 Introduction **1**
- 1.1 Multiple Independent Sources . . . 1
- 1.2 Bipartite Ramsey Graphs and 2-Source Dispersers . . . 2
- 1.3 (1-Source) Somewhere Condensers . . . 4
- 1.4 Some Related Subsequent Work . . . 4
- 1.5 Organization of the Paper . . . 5

2 Techniques and Overview of the Main Constructions **5**
- 2.1 A Somewhere Condenser with a Constant Number of Output Blocks . . . 5
- 2.2 A 2-source "Somewhere Extractor" . . . 6
- 2.3 A 4-source Extractor (and a 3-Source One) . . . 7
- 2.4 A 2-Source Disperser . . . 8
  - 2.4.1 Subsources and a high level outline . . . 9
  - 2.4.2 Nicely structured sources . . . 9
  - 2.4.3 The somewhere extractor `s_ext'`: extracting from nicely structured sources . . . 10
  - 2.4.4 Selecting using the "challenge-response mechanism" . . . 10
  - 2.4.5 The function `select` . . . 11
  - 2.4.6 Removing the unjustified assumption . . . 12

3 Preliminaries and Notations **13**
- 3.1 Probability Notations. . . . 13
  - 3.1.1 Sources, min-entropy, entropy rate and statistical distance . . . 13
  - 3.1.2 Conditioning random variables . . . 13
  - 3.1.3 Blocks . . . 14
- 3.2 Extractors and Dispersers . . . 14

4 Definitions of less standard concepts **14**
- 4.1 Subsources . . . 15
- 4.2 Somewhere$-\mathcal{P}$ sources . . . 16
- 4.3 Somewhere-uniform sources . . . 16

5 A Somewhere Condenser **17**
- 5.1 A basic condenser . . . 17
- 5.2 Composing condensers: Proof of Theorem 5.2 . . . 20

6 A 2-Source Somewhere-Extractor **21**
- 6.1 The construction . . . 22

7 A 3-Source Extractor **23**
- 7.1 An Appetizer: A 4-Source Extractor . . . 23
- 7.2 Down from 4 Sources to 3 . . . 24

8 A 2-Source Disperser **25**
- 8.1 The Construction . . . 26
  - 8.1.1 Partitions . . . 26
  - 8.1.2 Ingredients . . . 26
  - 8.1.3 Definition of the disperser `disp` . . . 27

8.2 An informal overview of the proof . . . 28
8.3 The Proof . . . 32
8.3.1 The main claim . . . 32
8.3.2 Nicely structured sources . . . 33
8.3.3 Extractor for nicely structured sources: proof of Claim 8.5 . . . 35
8.3.4 Rejecting incorrect partitions: proof of Claim 8.6 . . . 36
8.3.5 Existence of nicely structured sources: proof of Claim 8.7 . . . 37
8.3.6 Selecting the correct partition: proof of Claim 8.8 . . . 38

9 Proof for Stronger Notion of Dispersers **42**
9.1 Proof of Claim 9.1 . . . 42
9.2 Proof of Claim 9.2 . . . 43

10 Conclusion and open problems **45**

A Proof of Lemma 7.1 **48**

1 Introduction

Randomness extraction is the problem of distilling the entropy present in "weak random sources" into a useful, (nearly) uniform distribution. Its importance and wide applicability to diverse theoretical and practical areas of computer science has motivated a large body of research over the last 20 years.

The goal of this research is to design a "randomness extractor function" f such that when applying f on an input chosen according to certain probability distributions (that is on "weak random sources of randomness") one obtains outputs which are (statistically close) to uniformly distributed coin tosses. Much of this research assumes only that the given "weak source" has sufficient (min)-entropy[1], and that extractors can use an extra, short uniformly distributed independent "seed" to aid in distilling the randomness. Such a seed (of logarithmic length) is easily seen to be necessary in this setup. This line of research focuses on explicitly constructing extractors of small seed (and long output). The survey [Sha02] explains some of the recent developments in this area, and the papers [LRVW03, GUV09] contain the current state-of-art constructions in terms of the seed length and output length.

However, certain applications (especially in cryptography) cannot "afford" the use of an extra random seed (see e.g. [MP90, DS02, BST03]). To do without it, one must impose additional conditions (beyond entropy content) on the class of sources from which one extracts. This body of work, which includes [vN51, Blu86, SV84, CG88, OL89, Vaz87, CW89, CGH+85, TV00, MU01, KZ06, GRS04] considers a variety of restrictions, mostly structural, on the given sources, and shows how to extract from them deterministically (that is without an additional seed).

This paper provides several new explicit constructions of seedless extractors, dispersers and related objects (all defined below), significantly improving on previous results. We also give several weaker constructions, which are not quite seedless, but only use seeds of *constant size*. These are important as building blocks of the seedless ones, and some are interesting in their own right.

We now turn to describe some of the new constructions, and for each discuss history and related work. (We also discuss work that is subsequent to the publication of the conference version of this paper in Section 1.4). For the sake of brevity and clarity, we skip some of our results, and state others in less than full generality and precision.

1.1 Multiple Independent Sources

Background. Perhaps the most natural condition allowing seedless extraction is that instead of *one* source with high entropy, we have several independent ones. This model was suggested by Santha and Vazirani [SV84], and further studied by Chor and Goldreich [CG88] (who introduced the now standard notion of min-entropy).

A function $f : (\{0,1\}^n)^\ell \to \{0,1\}^m$ is called an ℓ-source extractor (with entropy requirement k and error parameter ϵ) if for every ℓ independent distributions $X_1, \cdots, X_\ell$, each with min-entropy k, the distribution $f(X_1, \cdots, X_\ell)$ (obtained by applying f on a sample from each source) has distance at most ϵ from the uniform distribution on $\{0,1\}^m$. The distance from the uniform distribution is measured according to statistical distance (see Section 3 for precise definitions).

For this model, the probabilistic method easily gives the best that can be hoped for: two sources and $\Theta(\log n)$ min-entropy suffice (and are necessary) for extraction. Moreover, such a function can be computed in time $2^{O(n^2)}$. While this is a far cry from the explicitness we want (that is we want

[1] A distribution X over $\{0,1\}^n$ has *min-entropy* (at least) k if for every $x \in \{0,1\}^n$, $\Pr[X = x] \leq 2^{-k}$. We say that X has *entropy rate* δ if X has min-entropy at least δn. A distribution X with rate δ is called a δ-source.

extractors computable in poly(n)-time), it will be used as a building block in our constructions. We call such a function $\texttt{opt}$ (for Optimal Extractor).

For two sources the best known explicit construction requires in contrast min-entropy $> n/2$.[2] The Hadamard function $\texttt{Had} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ defined by $\texttt{Had}(x,y) = \langle x, y\rangle \pmod 2$ was shown by [CG88] to be an extractor for two sources with rate slightly larger than $1/2$. Moreover, a natural variant $\texttt{Had}'$ which outputs $m = \Omega(n)$ was shown to be an extractor in [Vaz87] under the same conditions (see a simplified and improved analysis in [DEOR04]). The Hadamard function and its extension to long outputs $\texttt{Had}'$ will be an important building block in our construction. Constructing explicit extractors that work with sources with rate smaller than $1/2$ was open for many years even if one allows the extractor to use a large number of sources $\ell < n$.

Recently, [BIW04] (using the Sum-Product theorem of [BKT04]) gave an extractor that, for any $\delta > 0$ uses only a constant $\ell = \text{poly}(1/\delta)$ δ-sources.[3] Moreover, on n-bit sources, their extractor outputs n bits and is exponentially close to uniform. The analysis seems to require that the total entropy in all sources is at least the length of one source (and hence that $\ell > 1/\delta$). Still, this too will be a building block in our new constructions in which the number of sources is a constant independent of δ.

New Results. We construct a 3-source extractor which outputs a (nearly) unbiased bit (or any constant number of bits) for every entropy rate $\delta > 0$. That is, we prove the following theorem:

Theorem 1.1 (3-source extractor)**.** *For every constants $\delta, \epsilon > 0$ and $m \in \mathbb{N}$, and for every sufficiently large integer n there exists a* poly(n)*-time computable* 3*-source extractor* $\texttt{3ext} : \{0,1\}^{n\times 3} \to \{0,1\}^m$ *such that for every three independent δ-sources X_1, X_2, X_3, the distribution* $\texttt{3ext}(X_1, X_2, X_3)$ *is within ϵ statistical distance to the uniform distribution over* $\{0,1\}^m$*.*

We remark that there are subsequent works which can use a constant number of sources for lower rates and achieve better error. We discuss subsequent work in Section 1.4.

1.2 Bipartite Ramsey Graphs and 2-Source Dispersers

Ramsey Graphs. The probabilistic method was first used by Erdos to show the existence of *Ramsey graphs*: That is, a 2-coloring of the edges of the complete graph on N vertices such that no induced subgraph of size $K = (2 + o(1)) \log N$ is monochromatic. The best known explicit construction of such a coloring by [FW81] only achieves a much larger value: $K = 2^{\Theta(\sqrt{\log N \log\log N})}$.[4]

Bipartite Ramsey graphs. An even harder variant of this problem is the *bipartite Ramsey problem:* Construct a 2-coloring of the edges of the complete N by N bipartite graph such that no induced K by K subgraph is monochromatic. Setting $N = 2^n$, a coloring is a function $f : (\{0,1\}^n)^2 \to \{0,1\}$. It immediately follows that every 2-source extractor f with entropy-rate δ is a coloring for the bipartite Ramsey problem with $K = N^\delta$. We remark that it is easy to transform a bipartite Ramsey graph on $2N$ vertices into a (standard) Ramsey graph on N vertices while only multiplying K by two (thus explicit constructions of bipartite Ramsey graphs immediately translate into explicit constructions of Ramsey graphs). Until recently, the best known explicit

[2] In a subsequent work [Bou05] this was improved to min-entropy $(1/2 - \alpha)n$ for some constant $\alpha > 0$. We elaborate on subsequent work in Section 1.4.

[3] This extractor was proposed already in [Zuc90], but the analysis there relies on an unproven number theoretic assumption.

[4] In a subsequent work [BRSW06] extend the techniques developed in this paper and give an improved construction of Ramsey graphs. We discuss subsequent work in Section 1.4.

construction of bipartite Ramsey graphs was that implied by the aforementioned Hadamard 2-source extractor achieving $K > N^{1/2}$. Recently, a slight improvement to $K = N^{1/2}/2^{\sqrt{\log N}}$ (which in our terminology translates to $\delta = 1/2 - 1/\sqrt{n}$) was given by Pudlák and Rödl [PR04].[5] ‘

2-source dispersers. An equivalent formulation of this problem is constructing a 1-bit output 2-source *disperser* which is a well-known relaxation of an *extractor*. A 2-source (zero-error) *disperser* of entropy rate δ is a function $\texttt{disp} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$ such that for every 2 independent δ-sources X_1, X_2 we have that the support of $\texttt{disp}(X_1, X_2)$ is $\{0,1\}^m$. In words, every possible output occurs when the inputs range over all possible values in X_1, X_2 (and so only the support matters, not the individual probabilities in the input sources).[6] Note that when $m = 1$, a disperser is equivalent to a bipartite Ramsey graph with $K = N^\delta$.

New Results. We give an explicit construction of a 2-source disperser for any constant rate $\delta > 0$ and any constant output length m. Our disperser has the additional guarantee that every output is obtained with at least a constant probability, which depends only on δ and on the number of output bits. This construction can therefore be seen as being in between a disperser and an extractor. We prove the following theorem:

Theorem 1.2 (Two-source disperser)**.** *For every constants $\delta > 0$ and $m \in \mathbb{N}$, and for every sufficiently large integer n there exists a $\mathrm{poly}(n)$-time computable function $\texttt{disp} : \{0,1\}^{n\times 2} \to \{0,1\}^m$ such that for every two independent δ-sources X_1, X_2 over $\{0,1\}^n$, the support of $\texttt{disp}(X_1, X_2)$ is $\{0,1\}^m$. Moreover, there exists a constant $\vartheta = \vartheta(m, \delta) > 0$, such that for every $z \in \{0,1\}^m$,*

$$\Pr[\texttt{disp}(X_1, X_2) = z] \geq \vartheta(m, \delta) \ .$$

We immediately obtain the following corollary.

Corollary 1.3 (Bipartite Ramsey Graph)**.** *For every constant $\delta > 0$ there is a constant $\vartheta > 0$ and polynomial time algorithm such that for sufficiently large N the algorithm computes a 2-coloring c of the edges of the complete N by N bipartite graph $K_{N,N}$. (More precisely, when given an edge (x, y) the algorithm runs in time polynomial in the length of its input and produces a color $c(x, y)$). Furthermore, the coloring c has the property that every induced subgraph of size N^δ by N^δ has a ϑ proportion of edges of both colors (and in particular is not monochromatic).*

The result of Frankl and Wilson [FW81] gives explicit construction of Ramsey graphs with better parameters. We stress that [FW81] only achieves Ramsey graphs and not bipartite Ramsey graphs. In addition [FW81] only guarantees that no set of appropriate size is monochromatic. Therefore the aforementioned additional guarantee in our disperser can be seen as a qualitative improvement of the Frankl-Wilson construction for the case of Ramsey graphs.

We remark that there are subsequent works which extend the techniques of this paper and give a 2-source disperser for smaller min-entropy requirement which in turn translate to improved bipartite Ramsey graphs. More details are given in Section 1.4.

[5] The construction in that paper is only “weakly explicit” in the sense that the 2-coloring can be found in time polynomial in N. The Hadamard 2-source extractor (as well as all the constructions in this paper) are “strongly explicit” meaning that f is computable in time polynomial in $n = \log N$.

[6] In the extractor literature dispersers usually come with an error parameter ϵ, and then the requirement is that the output of f contains at least $(1 - \epsilon)$-fraction of all elements in $\{0,1\}^m$.

1.3 (1-Source) Somewhere Condensers

Intuitively, a *condenser* is a function whose output distribution is "denser" (has higher entropy rate) than its input distribution. Condensing can be viewed as a weaker form of extraction and indeed some constructions of extractors proceed by iterated condensing. Various condensers appear in [RR99, RSW00, RVW00, TSUZ07, LRVW03, CRVW02] and other works, mainly as building blocks to constructing extractors and expanders.

It is not hard to see that, like extractors, there are no deterministic condensers. However, unlike extractors, which require logarithmic seed, condensing is possible (for interesting parameters) with only constant length seed. As usual, this was shown via the probabilistic method, and no explicit construction was known. All constructions in the papers above either use a super-constant seed, or use a constant seed without guaranteeing the condensing property.[7]

New Results. We give the first explicit constant seed condenser for linear entropy. Loosely speaking, we show that for every $\delta > 0$ there are integers c, d and a poly(n)-time computable function $\texttt{con} : \{0,1\}^n \to (\{0,1\}^{n/c})^d$ (i.e. $\texttt{con}$ maps n bit strings into d blocks of length n/c), such that for every δ-source X there is at least one output block of $\texttt{con}(X)$ that is (exponentially close to) having entropy rate ≥ 0.9. Here the 0.9 is an arbitrary constant - we can get as close as we want to 1. (The precise statement appears in Section 5 and is slightly more technical - it allows the output distribution of $\texttt{con}$ to be a convex combination of distributions with the aforementioned property). We call these objects *somewhere condensers*.

As we shall see, this condenser is not only interesting in its own right, but it also serves as a basic block in our new constructions. Roughly speaking, it gives us the means to break the 1/2 rate barrier, as it converts an input source of rate *below* that barrier into one (of a few output blocks - something to be dealt with) whose rate is *above* that barrier.

Independently from our paper Raz [Raz05] gave a similar construction of a somewhere condenser (achieving the same parameters as our construction). Furthermore, using a new variant of the [LRVW03] merger, Raz constructs a condenser with the advantage that *most* of the output blocks are condensed.

1.4 Some Related Subsequent Work

In this section we survey related work that is subsequent to the publication of the conference version of this paper.

2-source extractors. The Hadamard extractor mentioned earlier was improved in two ways: Bourgain [Bou05] constructed an extractor for two δ-sources with rate $\delta < 1/2$ (more precisely, there exists a constant $\alpha > 0$ such that his extractor can work with sources of rate $\delta = (1/2 - \alpha)$). Raz [Raz05] constructed two source extractors where one of the sources is allowed to have very small entropy (logarithmic in n) while the other source requires rate $> 1/2$.

$O(1)$-source extractors. Rao [Rao06] constructed extractors which can use $O(\log n / \log k)$ sources with min-entropy k. Note that for $k = n^\alpha$ this gives extractors for $O(1/\alpha)$ sources. Barak et al. [BRSW06] gave an improved version of this extractor that achieves exponentially smaller error. Raz [Raz05] used some of the methodology developed in this paper as well as his aforementioned 2-source

[7] The latter are the so called "win-win" condensers introduced in [LRVW03] whose analysis shows that when they fail to condense, some other good thing must happen.

extractor to give 3-source extractors which improve over ours in the sense that two of the three sources are allowed to have logarithmic min-entropy.

2-source dispersers and bipartite Ramsey graphs. Pudlak [Pud06] gave a different (and simpler) construction of a coloring of the edges of the full bipartite N by N graph with 3 colors such that no induced subgraph of size $N^{1/2-\epsilon}$ by $N^{1/2-\epsilon}$ is monochromatic for some fixed small ϵ.

Gabizon and Shaltiel [GS08] improved the output length in our construction of 2-source dispersers. Their construction starts with the disperser of this paper and extends it to achieve an output length of $m = \Omega(n)$ where the hidden constant depends on δ.

Finally, building on the techniques introduced in this paper, Barak et al. [BRSW06] improved the main result of this paper and constructed a 2-source disperser for min-entropy $n^{o(1)}$. This translates into an improved construction of a bipartite Ramsey graph. In fact, this improved construction also improves the best previous construction of (non-bipartite) Ramsey graphs due to [FW81].

Condensers Zuckerman [Zuc06] gave another construction of a somewhere condenser with improved constants.

1.5 Organization of the Paper

In Section 2 we give a high level overview of our constructions. In Section 3 we give some preliminaries. In Section 4 we give formal definitions of less standard concepts that we use in the paper. The construction of the somewhere condenser is presented in Section 5. We use this construction to construct a new object which we call a "2-source somewhere extractor" in Section 6. This new object is used as a building block in our latter constructions. The construction of our 3-source extractor is presented in Section 7. The construction of our 2-source disperser is given in Section 8. In Section 9 we show that our 2-source disperser has the stronger property that any element in the output is obtained with constant probability. We present some conclusions and open problems in Section 10.

2 Techniques and Overview of the Main Constructions

In this section we give a high level overview of our constructions. The description is informal and we allow ourselves to oversimplify and ignore many technicalities in order to present the main ideas that are used in the constructions and proofs. The reader can skip this section at any time and move to the later sections for precise statements and proofs.

2.1 A Somewhere Condenser with a Constant Number of Output Blocks

The first object that we construct is a somewhere-condenser `con`. This is a function that receives an n-bits δ-source X and outputs d blocks each of length n/c (where c and d are constants that depend on δ) and has the guarantee that "one of the output blocks" is (close to) a 0.9-source.[8] This "final condenser" `con` is constructed by first constructing a "basic condenser" `bcon` in which the rate of the "good" output block is only guaranteed to improve slightly (namely to $(1 + \lambda(\delta)) \cdot \delta$ where $\lambda(\delta) > 0$ is an increasing function of δ). To obtain the final condenser `con` we apply `bcon` a constant number of times to condense from rate δ to rate 0.9.

[8] The precise definition is given in Section 5 and requires that there exists a random variable I (which may depend on X) such that $\texttt{con}(X)_I$ (namely, the I'th block of the output) is close to a 0.9-source. In this high level overview we oversimplify the presentation and say that "one of the output blocks" is (close to) a 0.9-source.

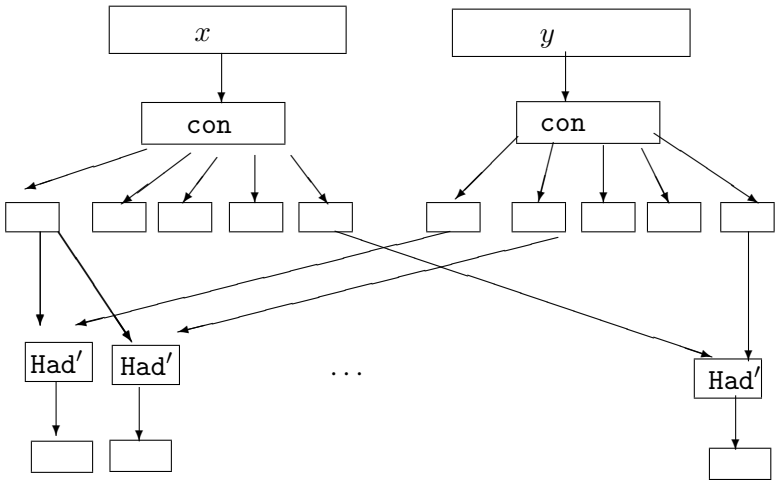

Figure 1: A 2-source somewhere extractor `s_ext`.

We now explain how to construct the basic condenser `bcon`. We make use of an extractor construction of Barak, Impagliazzo and Wigderson [BIW04] which gives that for every $\delta > 0$ there is a constant ℓ and an ℓ-source extractor `ext` which given ℓ n-bits δ-sources produces an n-bit source that is close to uniform. When given an n-bit source X, we partition it into ℓ blocks $X_1, \ldots, X_\ell$ each on strings of length $m = n/\ell$ and $\texttt{bcon}(X)$ outputs the blocks $X_1, \ldots, X_\ell$ and the additional block $\texttt{ext}(X_1, \ldots, X_\ell)$. Note that it is not necessarily the case that $X_1, \ldots, X_\ell$ are independent δ-sources. However, we argue that if none of the blocks $X_1, \ldots, X_\ell$ has rate larger significantly than δ then these blocks are "sufficiently independent" so that applying `ext` gives a source with rate significantly larger than δ.

To explain the intuition behind the proof note that if we measure Shannon entropy rather than min-entropy then by the chain rule for Shannon entropy we have that

$$\sum_{1 \le i \le \ell} H(X_i) \ge \sum_{1 \le i \le \ell} H(X_i | X_1, \ldots, X_{i-1}) = H(X) \ge \delta n = \ell \cdot (\delta m)$$

Therefore, if it isn't the case that there exists an i such that $H(X_i) > \delta m$ then for every i, $H(X_i) = \delta m$ and $H(X_i) = H(X_i | X_1, \ldots, X_{i-1})$ which implies that $X_1, \ldots, X_\ell$ are independent and we apply `ext` in a setting where it is guaranteed to extract.

The actual proof imitates the argument above but is more complex as min-entropy does not have a well behaved notion of conditional entropy. Furthermore, note that we want to condense to rate larger than $(1 + \lambda)\delta$ (as opposed to rate larger than δ). We show that if none of the X_i's has a significantly improved rate then if we compare the distribution of (the correlated variables) $X_1, \ldots, X_\ell$ to an experiment where we take samples from $X_1, \ldots, X_\ell$ in an independent way, then events that happen with small probability in the second experiment cannot have much larger probability in the first experiment.

2.2 A 2-source "Somewhere Extractor"

Our two main constructions in this paper are a 3-source extractor and a 2-source disperser. For both, an essential building block, is a 2-source somewhere extractor `s_ext` (short for "somewhere extractor") for linear entropy, which we describe next.

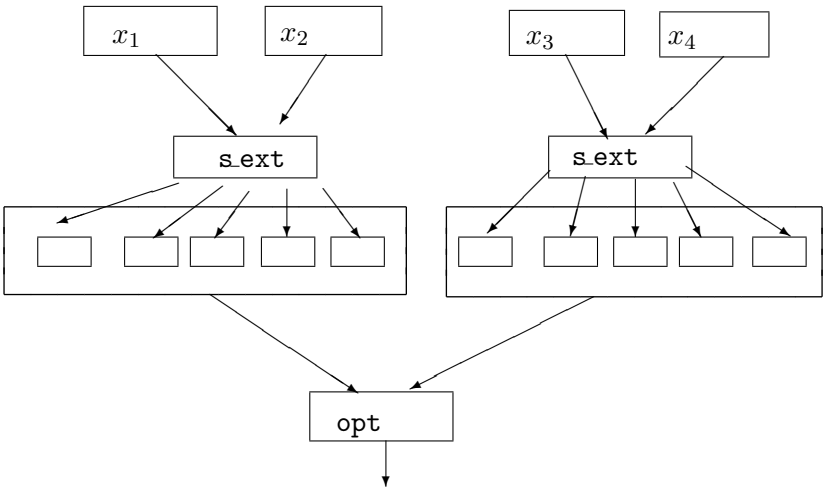

Figure 2: A 4-source extractor `4ext`.

What we prove is that for every $\delta > 0$ there are integers c, ℓ and a poly(n)-time computable function $\texttt{s_ext} : (\{0,1\}^n)^2 \to (\{0,1\}^{n/c})^\ell$, such that for every two independent δ-sources X_1, X_2 there is at least one output block $\texttt{s_ext}(X_1, X_2)_i$ which is (exponentially close to) uniform.

Constructing the somewhere extractor `s_ext` is simple, given the condenser `con` of the previous subsection. To compute $\texttt{s_ext}(X_1, X_2)$, compute the output blocks of $\texttt{con}(X_1)$ and $\texttt{con}(X_2)$. By definition, some output block of each has rate $> .9$. We don't know which, but we can try all pairs! For each pair we compute the aforementioned Vazirani variant $\texttt{Had}'$ of the Hadamard 2-source extractor for rate $> 1/2$ [Vaz87] to obtain a constant number of linear length blocks, one of which is exponentially close to uniform. Formally, if d is the number of output block of `con`, then `s_ext` will produce $\ell = d^2$ blocks, with $\texttt{s_ext}(X_1, X_2)_{(i,j)} = \texttt{Had}'(\texttt{con}(X_1)_i, \texttt{con}(X_2)_j)$. This construction is depicted in Figure 1.

To see the power of this gadget, let us first see (intuitively) how to get from it a *deterministic* 4-source extractor for linear entropy. Later we will employ it in several ways to get our 2-source disperser.

2.3 A 4-source Extractor (and a 3-Source One)

In this subsection we explain how to construct a 4-source extractor `4ext`, and then how to modify it to the promised 3-source extractor `3ext`. These will combine the 2-source somewhere extractor `s_ext` with the aforementioned nonexplicit optimal 2-source extractor `opt`.

Recall that our 2-source *somewhere* extractor `s_ext` produces a constant number (say) ℓ of linear length output blocks, one of which is random. First we note that we can w.l.o.g. assume that `s_ext` produces ℓ shorter blocks with the same guarantee (as a prefix of a random string is random).

Let us indeed output only a constant b bits in every block (satisfying $b \geq \log(\ell b)$). Concatenating all output blocks of this $\texttt{s_ext}(X_1, X_2)$ gives us a distribution (say Z_1) on ℓb bits with min-entropy $\geq b$. If we have 4 sources, we can get another independent such distribution Z_2 from $\texttt{s_ext}(X_3, X_4)$. But note that these are two independent distributions on a constant number of bits with sufficient min-entropy for (existential) 2-source extraction. Now apply an optimal (nonexplicit) 2-source extractor on Z_1, Z_2 to get a uniform bit; as ℓb is only a constant, such an extractor can be found

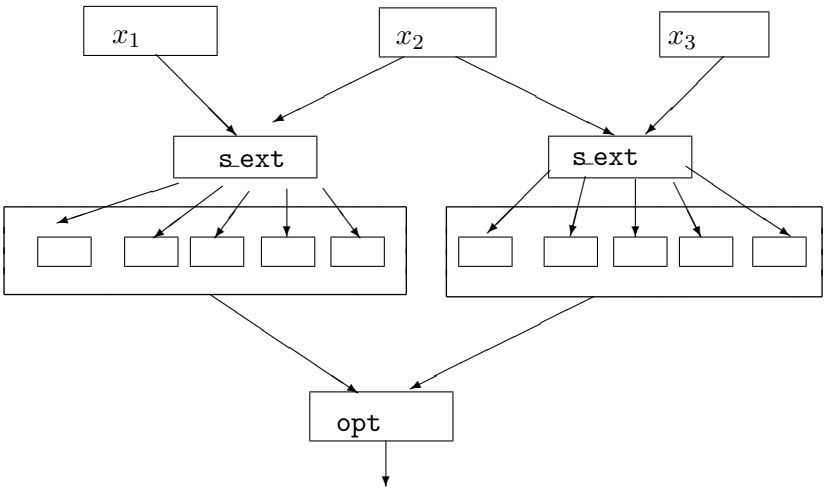

Figure 3: A 3-source extractor `3ext`.

in constant time by brute-force search! To sum up, our 4-source extractor is

$$\texttt{4ext}((X_1, X_2); (X_3, X_4)) = \texttt{opt}(\texttt{s_ext}(X_1, X_2), \texttt{s_ext}(X_3, X_4))$$

See Figure 2 for a schematic description of this construction.

Reducing the number of sources to 3 illustrates a simple idea we'll need later. We note that essentially all 2-source constructions mentioned above are "strong". This term, borrowed from the seeded extractor literature, means that the output property is guaranteed for almost every way of fixing the value of *one* of the two input sources. With this in mind, we can *reuse* (say) X_2 in the second somewhere extractor `s_ext` instead of X_4 to yield a 3-source extractor

$$\texttt{3ext}((X_1, X_2, X_3) = \texttt{opt}(\texttt{s_ext}(X_1, X_2), \texttt{s_ext}(X_3, X_2))$$

This construction is depicted in Figure 3. A randomly chosen sample x_2 from X_2 will w.h.p have the property that both $\texttt{s_ext}(X_1, x_2)$ and $\texttt{s_ext}(X_3, x_2)$ are somewhere random. Note that once X_2 is fixed to a value x_2 the two aforementioned distributions are independent. It follows that for most fixed values of X_2 we apply `opt` on independent distributions with sufficient entropy and we indeed obtain an output that is (close to) uniform.

2.4 A 2-Source Disperser

In this subsection we give a high level overview of our construction of a 2-source disperser. The construction is significantly more complicated than those of the previous objects and relies on the somewhere extractor `s_ext` and the nonuniform optimal 2-source extractor `opt` that we explained previously.

An appealing approach to construct a 2-source disperser (or a 2-source extractor) is to try and select the "correct output block" of a somewhere random extractor. That is, when given two inputs x, y that are sampled from independent δ-sources X, Y one can run $\texttt{s_ext}(x, y)$ to obtain d output blocks where one of them is random and then try to select the correct output block. An obvious obstacle is that the only information available to the disperser at this point is the pair of inputs x, y and this doesn't suffice to determine the correct block.

An important contribution of this paper is showing that this naive idea can be implemented in some setups. To explain how the naive idea can make sense we need the notion of subsources that is explained next.

2.4.1 Subsources and a high level outline

We say that an n-bit source X' is a subsource of an n-bit source X if there exists a function $f : \{0,1\}^n \to \{0,1\}$ such that $\Pr[f(X) = 1] > 0$ and $X' = (X|f(X) = 1)$ (that is the conditional distribution of X when conditioned on the event $\{f(X) = 1\}$). We call $\{f(X) = 1\}$ a defining event for X'. We construct the disperser by designing two procedures:

- somewhere extractor $\texttt{s_ext}'(x, y)$ (with roughly the same parameters as that of $\texttt{s_ext}$).
- A function $\texttt{select}(x, y)$ which when given x, y outputs a name of an output block of $\texttt{s_ext}'$.

We will show that: For any two independent δ-sources X, Y there exists a subsource $\tilde{X}$ of X and a subsource $\tilde{Y}$ of Y (note that $\tilde{X}, \tilde{Y}$ are independent by definition) such that:

- There is a constant i such that $\texttt{select}(\tilde{X}, \tilde{Y})$ outputs i with probability very close to one.
- $\texttt{s_ext}'(\tilde{X}, \tilde{Y})_i$ is (very close to) uniform and therefore it outputs every output string $z \in \{0,1\}^m$ with positive probability.

Together, these two properties give that $\texttt{disp}(x, y) = \texttt{s_ext}(x, y)_{\texttt{select}(x,y)}$ is a 2-source disperser. This is because $\texttt{disp}$ indeed outputs any possible output string z with positive probability when applied on $\tilde{X}, \tilde{Y}$. Furthermore, the defining events $\{f_1(X) = 1\}$ and $\{f_2(Y) = 1\}$ of the subsources $\tilde{X}, \tilde{Y}$ both have positive probability and are independent. It follows that when we sample from (X, Y) we have positive probability to land in $\{f_1(X) = 1 \cap f_2(Y) = 1\}$ and when this happens we output z with positive probability.

2.4.2 Nicely structured sources

Before explaining how to construct the procedures $\texttt{s_ext}'$ and $\texttt{select}$ we observe that any δ-source X has a subsource X' that is "nicely structured". (The "good subsource" $\tilde{X}$ guaranteed above is going to be a subsource of X' which in addition to being nicely structured will have additional properties). We now explain what we mean by a nicely structured source. Given an n bit source X we partition it into $t = 10/\delta$ blocks (each of length $\delta n/10$ bits) which we denote by $X_1, \ldots, X_t$. When given an $i \in [t]$ we consider three blocks: $X_{<i}$ (the concatenation of $X_1, \ldots, X_{i-1}$), X_i (the i'th block) and $X_{>i}$ (the concatenation of $X_{i+1}, \ldots, X_t$). We say that a source X is *nicely structured* according to $i \in [t]$ if $X_{>i}$ is fixed to some constant value, X_i has constant rate and $X_{<i}$ has constant rate when conditioned on X_i. It is not hard to show that indeed every δ-source has a nicely structured subsource X'. In order to explain this argument at a high level let us replace min-entropy with Shannon entropy (the actual proof uses a similar strategy but is more complicated as min-entropy does not have well behaved notion of conditional entropy). By the chain rule for entropy we have that:

$$\delta n \leq H(X) = \sum_{i \in [t]} H(X_i | X_{i+1}, \ldots, X_t)$$

Let i be the largest index such that $H(X_i | X_{i+1}, \ldots, X_t) \geq (\delta/2) \cdot (n/t)$ (in words that the conditional rate of the i'th block is larger than $\delta/2$). Note that such an index must exist. Furthermore,

note that $H(X_{<i}|X_i, X_{>i}) \geq \delta n - (\delta n/2 + \delta n/10) \geq \delta n/4$. This is because by the way we chose i, $H(X_{>i}) < \delta n/2$ and we also have that $H(X_i) \leq |X_i| = \delta n/10$ and so the remaining entropy must lie in $X_{<i}$. Let X' be the subsource $(X|X_{>i} = z)$ for some "typical string" z and note that X' is nicely structured.

2.4.3 The somewhere extractor s_ext′: extracting from nicely structured sources

It turns out that we have already constructed an extractor for two independent nicely structured sources! We can show that given two independent sources $\tilde{X}, \tilde{Y}$ that are nicely structured according to (i_1, i_2) respectively, applying our 4-source extractor $\texttt{4ext}(\tilde{X}_{<i_1}, \tilde{Y}_{i_2}, \tilde{Y}_{<i_2}, \tilde{X}_{i_1})$ produces a uniform output. This is somewhat surprising as the four distributions above are not independent. Still, we show that they are sufficiently independent for the argument showing the correctness of `4ext` to go through. We will not explain why this is the case in this high level outline. Let us just mention that the reasoning is similar in spirit to the case of the analysis of `3ext` and once again the "strongness property" of `s_ext` plays an important role in the argument.

We now explain how to design the aforementioned somewhere extractor `s_ext′`. This extractor will have t^2 output blocks where we think of each block $i \in [t^2]$ as a pair $(i_1, i_2) \in [t]^2$. We define $\texttt{s_ext}'(x, y)$ to be the output of `4ext` when partitioning x, y according to (i_1, i_2). Note that we indeed have that for any two nicely structured sources $\tilde{X}, \tilde{Y}$ there exists an $i = (i_1, i_2)$ such that $\texttt{s_ext}(\tilde{X}, \tilde{Y})_i$ produces a uniform output.

It may not be clear at this point why the construction of `s_ext′` gives progress over that of `s_ext`. After all, `s_ext` is a somewhere extractor that works on any sources (not necessarily nicely structured). The main advantage is that we will be able to couple `s_ext′` with a function `select` that will find the right output block $i = (i_1, i_2)$ when applied on nicely structured sources.

2.4.4 Selecting using the "challenge-response mechanism"

Consider a source X that is nicely structured according to an index $i_1 \in [t]$. Our goal is to show that we can *find* the index i_1. (We remark that in the final argument X will be a nicely structured subsource of the original source).
The following properties of i_1 will allow us to find it:

- For $i \geq i_1$, $X_{>i}$ is fixed (as $X_{>i_1}$ is fixed).
- For $i < i_1$, $X_{>i}$ has constant rate (as $X_{>i}$ contains X_{i_1} as a substring, and X_{i_1} has constant rate).

Thus to "select" i_1 we need a test that will distinguish the case that $X_{>i}$ is fixed from the case when $X_{>i}$ has constant rate.

An important contribution of this paper is developing a methodology (which we call "the challenge-response mechanism") that gives such a test. The construction and analysis of the test are quite involved and require many details. A precise description of the construction appears in Section 8.1 and a detailed high level informal overview of the analysis appears in Section 8.2.

In the remainder of this informal overview we attempt to present the high level ideas that come into the challenge-response mechanism (without presenting the precise construction). In order to do this we make an unjustified simplifying assumption and assume the existence of a very strong object (stronger than the disperser that we are constructing, and in fact so strong that it doesn't even exist). We explain how to select the "right block" using this object. We later comment how one can replace the simplifying assumption with the components that are available to us.

Unjustified simplifying assumption: A 1-source somewhere extractor: We assume that for any $\delta > 0$ there exist integers ℓ, c and a $poly(n)$-time computable function $\texttt{1ext} : \{0,1\}^n \to (\{0,1\}^{n/c})^\ell$ such that for any δ-source X there exists $i \in [\ell]$ such that $\texttt{1ext}(X)_i$ is uniformly distributed.

It is easy to show that such an object does not exist. Still we now explain how using this object one can design a function $\texttt{select}(x)$ that "selects" the correct index i_1 (in a sense to be explained precisely below) when applied on a nicely structured source. In the actual proof we design a more complicated function `select` that is applied on samples x, y from the two independent sources. Jumping ahead we mention that instead of `1ext` we will use the somewhere extractor `s_ext` (that indeed produces a somewhere random distribution when applied on two independent constant rate sources).

2.4.5 The function `select`

Let X' be a nicely structured source according to index i_1. Recall that we are assuming the existence of the 1-source somewhere extractor `1ext`. We design a function $\texttt{select}(x)$ with the following property: There exists a subsource $\tilde{X}$ of X' that is nicely structured according to i_1 such that $\texttt{select}(\tilde{X})$ outputs i_1 with probability one.

Let us first observe why this is good enough for our purposes. When given a δ-source X we already showed that it has a nicely structured subsource X' and we now show that this subsource has a nicely structured subsource $\tilde{X}$ on which `select` chooses the "correct index" i_1 with probability close to one. Therefore, this subsource $\tilde{X}$ has the two properties listed in Section 2.4.1 needed for our construction to succeed.
We now explain how to implement the function $\texttt{select}(x)$:

Challenge strings: For every $i \in [t]$ we compute $c_i(x) = \texttt{1ext}(x_{>i})$ (we call these strings "challenge strings"). In this application of `1ext` we output ℓ blocks of length k for some large constant k. It follows that the overall length of challenge strings is some constant ℓk.

Response strings: We compute $r(x) = \texttt{1ext}(x)$. In this application of `1ext` we output ℓ blocks of length ℓk (that is the length of challenge strings). In other words, $r(x)$ is composed of a constant number of blocks $r_1(x), \ldots, r_\ell(x)$ (which we call response strings) where each one is of length of a challenge string.

We say that "the i'th challenge is responded" if there exists a response string r_v such that $r_v = c_i$. The function `select` outputs the minimal i such that the i'th challenge is responded (if there exists such an i). We first observe the following properties of challenge strings and response strings:

Constant rate block: If X is a source such that $X_{>i}$ has constant rate then for any v:
$\Pr[r_v(X) = c_i(X)] \leq 2^{-k}$.

Fixed block: If X is a source such that $X_{>i}$ is fixed then there exists a v such that:
$\Pr[r_v(X) = c_i(X)] \geq \rho$ where $\rho = 2^{-\ell k} < 2^{-k}$.

Before explaining why these two properties hold, let us explain why they are useful. This is not obvious at all. It would have made more sense if we could guarantee that ρ is very close to one. In that case, the event we are considering distinguishes between the two cases and intuitively this can help us find out which of the two cases happened. However, the way the two items are stated, the

two properties above do not seem helpful as they do not distinguish between the two cases. (For example, it can be the case that the probability is 2^{-k} in both cases).

In order to use the properties above we note that $X'_{>i_1}$ is fixed and by the second property (the fixed block case) there exists a v_1 such that $\Pr[r_{v_1}(X') = c_{i_1}(X')] \geq \rho$. Let us consider the subsource:

$$\tilde{X} = (X' | r_{v_1}(X') = c_{i_1}(X'))$$

Note that in this subsource the challenge c_{i_1} is responded with probability one! Furthermore as ρ is a constant the conditioning above reduces the entropy of X' by a constant number of bits. This means that $\tilde{X}$ has roughly the same entropy as X' and is therefore also nicely structured. We already observed that this gives that for any $i < i_1$, $\tilde{X}_{>i}$ has constant rate and so by the first property above (the constant rate case) for any v, $\Pr[r_v(\tilde{X}) = c_i(\tilde{X})] \leq 2^{-k}$. We can choose the constant k large enough to do a union bound over all response strings and it follows that w.h.p. the i'th challenge string is not responded. We can proceed and do a union bound over all $i < i_1$ and conclude that w.h.p. on all these i's the i'th challenge is not responded which means that $\texttt{select}(\tilde{X}) = i_1$ as required! Note that we indeed have that $\tilde{X}$ is a subsource of X which meets all our requirements.

Finally, let us verify that the two properties above hold: For the constant rate block case note that for any value r of any response string r_v we have that conditioning on the event $\{r_v(X) = r\}$ reduces the entropy by at most the length of the response string which is constant. Thus, $X_{>i}$ still retains constant rate and so $c_i(X) = \texttt{1ext}(X_{>i})$ has an output block of length k that is uniformly distributed. It follows that for any response string r_v, $\Pr[r_v(X) = c_i(X)] \leq 2^{-k}$.

For the fixed block case note that if $X_{>i}$ is fixed then $c_i(X) = \texttt{1ext}(X_{>i})$ is fixed. On the other hand there exists a v such that $r_v(X) = \texttt{1ext}(X)_v$ is uniformly distributed. As the length of the challenge string is a constant ℓk we have that $\Pr[r_v(X) = c_i(X)] \geq 2^{-\ell k}$ and indeed $\rho = 2^{-\ell k}$.

2.4.6 Removing the unjustified assumption

Let us briefly review the argument so far. We designed a procedure `select` and showed that for any δ-source X there exists a subsource $\tilde{X}$ which is nicely structured according to i_1 and furthermore, w.h.p. $\texttt{select}(\tilde{X})$ selects the index i_1. When given two independent δ-sources X, Y we consider their "good subsources" $\tilde{X}, \tilde{Y}$. On these subsources we can select the "correct indices" (i_1, i_2) and once we have these indices we can apply the somewhere extractor $\texttt{s_ext}'(\tilde{X}, \tilde{Y})_{(i_1,i_2)}$ to produce a uniformly distributed output.

However, we have assumed the existence of `1ext` that enabled us to construct the procedure `select`. It is easy to show that there does not exist a 1-source somewhere extractor with a small number of output blocks. Nevertheless, it is important to observe that we already constructed a 2-source somewhere extractor `s_ext` with the same guarantees on the output as that of `1ext`! When trying to select the index i_1 of the first source X we will use the fact that we have another independent source Y at our disposal. In the actual construction we replace invocations of $\texttt{1ext}(\cdot)$ by $\texttt{s_ext}(\cdot, y)$.

This introduces many problems where the most serious one is that in several places in the argument we considered conditioning a source X on the event of fixing one response string (that is an event of the form $\{\texttt{1ext}(X)_v = r\}$ for some constants v, r). In the actual analysis this will be replaced by conditioning two independent sources X, Y on an event of the form $\{\texttt{s_ext}(X, Y)_v = r\}$. However, this conditioning may make the two sources dependent and the argument does not go through. In order to solve this problem we use a more complicated construction of response strings: On a very high level we will make sure that response strings are generated by applying `s_ext` on very short substrings of x, y. Using this choice we can show that even after conditioning X, Y on

the event "the response string r_v is fixed to a value r" the two sources are sufficiently independent for the argument to go through.

The precise construction is given in Section 8.1. We also provide an overview of the analysis in Section 8.2.

3 Preliminaries and Notations

In this section we give some standard preliminaries and formal definitions of extractor and dispersers.

3.1 Probability Notations.

3.1.1 Sources, min-entropy, entropy rate and statistical distance

We will usually deal with random variables which take values in $\{0,1\}^n$. 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 be $\min_x\{-\log_2(\Pr[X=x])\}$, or equivalently $\log_2(1/\max_x\{\Pr[X=x]\})$. We shall usually identify a random variable X with the distribution it induces. The *entropy rate* of an n-bit source X is defined to be $H^\infty(X)/n$. A δ-source is a distribution with entropy rate at least δ. The support of a random variable X, denoted by $\mathsf{Supp}(X)$, is the set of elements x for which $\Pr[X=x]>0$. If $\Pr[X=x]=\Pr[X=x']$ for every $x,x'\in\mathsf{Supp}(X)$ we say that X is a flat source.

Let X and Y be random variables taking values in a set Λ. The *statistical distance between X and Y*, denoted by $\mathsf{dist}(X,Y)$ is defined to be $\frac{1}{2}\sum_{x\in\Lambda}|\Pr[X=x]-\Pr[Y=x]|$. We say that X is ϵ-close to Y if $\mathsf{dist}(X,Y)\le\epsilon$.

3.1.2 Conditioning random variables

Given a random variable X and an event E such that $\Pr[E]>0$ we use $(X|E)$ to denote the probability distribution on values of X that is obtained when conditioning the probability space on the event E. More precisely the distribution $(X|E)$ assigns probability $\Pr[X=x|E]$ to any $x\in\mathsf{Supp}(X)$. We need the following standard lemmas (we include the proofs for completeness):

Lemma 3.1. *Let X be a random variable such that $H^\infty(X)\ge k$, and let E be an event such that $\Pr[E]\ge p$ and let X' be the distribution $(X|E)$. Then, $H^\infty(X')\ge k-\log(1/p)$.*

Proof. For any x in the support of X we have that:

$$\Pr[X=x|E]=\frac{\Pr[X=x\wedge E]}{\Pr[E]}\le\frac{\Pr[X=x]}{p}\le 2^{-(k-\log(1/p))}$$

□

Lemma 3.2. *Let X_1,X_2 be random variables such that $H^\infty((X_1,X_2))\ge k$ and X_2 takes values in $\{0,1\}^r$. Then $H^\infty(X_1)\ge k-r$.*

Proof. By an averaging argument for every x_1 in the support of X_1 there exists x_2 in the support of X_2 such that $\Pr[X_2=x_2|X_1=x_1]\ge 2^{-r}$. Note that:

$$\Pr[X_1=x_1]\cdot\Pr[X_2=x_2|X_1=x_1]=\Pr[X_1=x_1\wedge X_2=x_2]\le 2^{-k}$$

It follows that: $\Pr[X_1=x_1]\le 2^{-k}/2^{-r}=2^{-(k-r)}$. □

Lemma 3.3. *Let X_1, X_2 be random variables such that $H^\infty(X_1) \geq k$ and X_2 takes values in $\{0,1\}^r$. Then for every $\rho > 0$ with probability $1-\rho$ over choosing $x_2 \leftarrow X_2$ we have that $H^\infty(X_1|X_2 = x_2) \geq k - r - \log(1/\rho)$.*

Proof. Let $G = \{x_2 : \Pr[X_2 = x_2] \geq 2^{-(r+\log(1/\rho))}\}$. Note that $\Pr[X_2 \notin G] \leq 2^r \cdot 2^{-(r+\log(1/\rho))} \leq \rho$. For $x_2 \in G$ we can use Lemma 3.1 on the variable X_1 and the event $\{X_2 = x_2\}$ to show that $H^\infty(X_1|X_2 = x_2) \geq k - r - \log(1/\rho)$. □

3.1.3 Blocks

Let X be an n-bit source, and let us write it as a vector of one-bit variables $(X_1, \ldots, X_n)$. For a set $S \subseteq [n]$ of coordinates we define the X_S, to be $|S|$-bit source $(X_i)_{i \in S}$, obtained from X by only taking its S coordinates. We note that by Lemma 3.2 for every n-bit source X and $S \subseteq [n]$ it holds that $H^\infty(X_S) \geq H^\infty(X) - (n - |S|)$. Also, if $\mathsf{dist}(X, Y) < \epsilon$ then $\mathsf{dist}\,(X_S, Y_S) < \epsilon$ for every $S \subseteq [n]$.

3.2 Extractors and Dispersers

In this section we define some of the objects we will later construct, namely multiple-source extractors and dispersers.

Definition 3.4 (Multiple-source extractor)**.** A function $\mathtt{ext} : \{0,1\}^{n \times \ell} \to \{0,1\}^m$ is called an *ℓ-source extractor with entropy requirement k and error ϵ* if for every independent n-bit sources $X^{(1)}, \ldots, X^{(\ell)}$ satisfying $H^\infty(X^{(i)}) \geq k$ for $i = 1, \ldots, \ell$ it holds that

$$\mathsf{dist}\Big(\mathtt{ext}(X^{(1)}, \ldots, X^{(\ell)}), U_m\Big) \leq \epsilon$$

Definition 3.5 (Multiple-source disperser)**.** A function $\mathtt{disp} : \{0,1\}^{n \times \ell} \to \{0,1\}^m$ is called an *ℓ-source disperser with entropy requirement k and error ϵ* if for every independent n-bit sources $X^{(1)}, \ldots, X^{(\ell)}$ satisfying $H^\infty(X^{(i)}) \geq k$ for $i = 1, \ldots, \ell$ it holds that

$$|\mathsf{Supp}(\mathtt{disp}(X^{(1)}, \ldots, X^{(\ell)}))| \geq (1-\epsilon)2^m$$

The disperser is *errorless* if $\epsilon = 0$ (i.e., if $|\mathsf{Supp}(\mathtt{disp}(X^{(1)}, \ldots, X^{(\ell)}))| = 2^m$).

Note that a multiple-source extractor is always a multiple-source disperser with the same parameters. Generally speaking, when constructing extractors and dispersers our goal is on one hand to use the weakest assumptions on the input and hence we want to minimize the number of samples ℓ and the min-entropy requirement k. On the other hand we want to obtain the strongest guarantees on the output, and thus we want to maximize the output length m and minimize the error ϵ.

We note that one can also define *asymmetric* variants of extractors and dispersers, in which the min-entropy requirements and the input length differ for each of the source.
In this paper we use the following construction by Barak, Impagliazzo and Wigderson [BIW04].

Theorem 3.6 ([BIW04])**.** *There exists $\gamma > 0$ such that for every $\delta > 0$ there exists $\ell = (\frac{1}{\delta})^{O(1)}$ and a polynomial time computable ℓ-source extractor $\mathtt{ext} : \{0,1\}^{n \times \ell} \to \{0,1\}^n$ with entropy requirement $k = \delta n$ and error $\epsilon = 2^{-\gamma n}$.*

4 Definitions of less standard concepts

In this section we introduce two key concepts that are used in the paper.

4.1 Subsources

As explained in the introduction, a key notion in our construction of 2-source dispersers is that of subsources.

Definition 4.1 (Subsource)**.** Let X be an n-bit source in some probability space. We say that an event A is determined by X if there exists a function $f : \{0,1\}^n \to \{0,1\}$ such that $A = \{f(X) = 1\}$.

Let X and X' be n-bit sources. For $0 < \alpha \leq 1$, we say that X' is a subsource of X of measure α if there exists an event A that is determined by X such that $\Pr[X \in A] \geq \alpha$ and $X' = (X|A)$. We call A a defining event for X'. We say that X' is a subsource of X if there exists $\alpha > 0$ such that X' is a subsource of X of measure α.

The following fact is obvious from the definition.

Fact 4.2. *Let X be an n-bit source. Let X' be a subsource of X of measure α_1, and X'' be a subsource of X' of measure α_2 then X'' is a subsource of X of measure $\alpha_1 \cdot \alpha_2$.*

In a typical setting our probability space consists of two independent n-bit δ-sources X, Y. It is important to notice that if A is some event in this probability space it does not necessarily follows that $(X|A)$ is a subsource of X. This is because only events that are determined by X are allowed. The following trivial fact will be useful.

Fact 4.3. *Let X, Y be two independent sources and let X' be a subsource of X of measure α_1 and Y' be a subsource of Y of measure α_2 then the distribution (X', Y') (in which the two variables are independent) is a subsource of measure $\alpha_1 \cdot \alpha_2$ of the distribution (X, Y)*

Subsource collection. Given independent n-bits sources X, Y we also define a notion of a *collection of subsources*. This notion is only used in the proof that our 2-source disperser has the additional property that it hits every element in its output range with constant probability. (This proof is given in Section 9). The reader can safely skip this notion at a first reading and move to the next subsection.

Definition 4.4 (Subsource collection)**.** Let X, Y be independent n-bit sources. For $0 < \alpha \leq 1$, we say that subsources $X_1, \ldots, X_k$ of X and $Y_1, \ldots, Y_k$ of Y are a subsource collection of (X, Y) of measure α if there are events $A_1, \ldots, A_k$ defining $X_1, \ldots, X_k$ and $B_1, \ldots, B_k$ defining $Y_1, \ldots, Y_k$ such that:

- For every $i \neq i'$, $(A_i \times B_i) \cap (A_{i'} \times B_{i'}) = \emptyset$.
- $\sum_{i \in [k]} \Pr[X \in A_i] \Pr[Y \in B_i] \geq \alpha$.

Note that by Fact 4.3 the subsources X', Y' in Fact 4.3 constitute a subsource collection of (X, Y) of measure $\alpha_1 \cdot \alpha_2$. The advantage of subsource collections is that we are sometimes able to show the existence of a "useful" subsource collection of (X, Y) with "large" measure in cases where we do not have "useful" subsources X' of X and Y' of Y with large measure.
Continuing the analogy between subsources and subsource collections, we also state the following analog of Fact 4.2.

Fact 4.5. *Let X, Y be independent n-bit sources and let w_1, w_2 be integers. Let $X'_1, \ldots, X'_{w_1}$ and $Y'_1, \ldots, Y'_{w_1}$ be a subsource collection of (X, Y) of measure α_1. Furthermore, for every $j \in [w_1]$ let $X''_{j,1}, \ldots, X''_{j,w_2}$ and $Y''_{j,1}, \ldots, Y''_{j,w_2}$ be a subsource collection of (X'_j, Y'_j) of measure α_2. Then, the collection $(X''_{j_1,j_2})_{j_1 \in [w_1], j_2 \in [w_2]}$ and $(Y''_{j_1,j_2})_{j_1 \in [w_1], j_2 \in [w_2]}$ is a subsource collection of (X, Y) of measure $\alpha_1 \cdot \alpha_2$.*

4.2 Somewhere$-\mathcal{P}$ sources

We discuss families or *properties* of random variables. Let $\mathcal{P}$ be a property of sources (namely, $\mathcal{P}$ is a set of all distributions that have the property). We say that a random variable X is ϵ-close to having the property $\mathcal{P}$, if there exists a random variable $Y \in \mathcal{P}$ that is ϵ-close to X. Let $\mathcal{P}_\epsilon$ denote the property of being ϵ-close to $\mathcal{P}$.

We now define the notion of *somewhere-$\mathcal{P}$ sources*, which intuitively means that a source X has the property $\mathcal{P}$ in one of its blocks.

Definition 4.6 (Somewhere$-\mathcal{P}$ sources)**.** Let $\mathcal{P}$ be a property of n-bit sources. Let X be an $(n \times \ell)$-bit source, and let us regard it as a concatenation of ℓ consecutive blocks, $X = (X_1, \ldots, X_\ell)$, where each X_i is of length n. A selector I for X is a random variable (that may depend on X) taking values in $[\ell]$.

X is somewhere-$\mathcal{P}$ if there exists a selector I such that the n-bit source X_I has the property $\mathcal{P}$. We use this notion only when the partition of X into sub-blocks is clear from the context.

We observe that being ϵ-close to somewhere-$\mathcal{P}$ is equivalent to being somewhere-$\mathcal{P}_\epsilon$.

Lemma 4.7. *Let $\mathcal{P}$ be a property of m-bit sources an let X be a distribution over $\{0,1\}^{m \times \ell}$. Then, X is ϵ-close to somewhere-$\mathcal{P}$ if and only if X is somewhere-$\mathcal{P}_\epsilon$.*

Proof. Let X be ϵ-close to somewhere-$\mathcal{P}$. That is, there are random variables Y and I such that X is ϵ-close to Y and Y_I has the property $\mathcal{P}$. W.l.o.g. we can imagine that the probability space of Y, I is composed of two independent random variables Y, R and that $I = f(Y, R)$ for some function f. We now consider the probability space which consists of random variables X', R where X' is distributed according to X and R is independent of X'. Let $I' = f(X', R)$. It follows that $X'_{I'}$ is ϵ-close to Y_I and thus, X is somewhere-$\mathcal{P}_\epsilon$.

For the other direction, we use the fact that for any two distributions P, Q that are ϵ-close there exist (correlated) random variables P', Q' such that P' is distributed according to P, Q' is distributed according to Q and $\Pr[P' \neq Q'] \leq \epsilon$. Let X be somewhere-$\mathcal{P}_\epsilon$. It follows that there exists a correlated random variable I such that X_I is ϵ-close to a distribution in $\mathcal{P}$. By the aforementioned fact we can add to the probability space of X, I a correlated random variable Z that has property $\mathcal{P}$ and $\Pr[X_I \neq Z] \leq \epsilon$. We now define a random variable Y over $\{0,1\}^{m \times \ell}$ as follows: we set $Y_J = X_J$ for every $J \neq I$ and set $Y_I = Z$. It follows that Y is ϵ-close to X and note that Y is somewhere-$\mathcal{P}$ using the selector I. $\square$

4.3 Somewhere-uniform sources

We are especially interested in the case where the property $\mathcal{P}$ is "uniform". More precisely, a somewhere-uniform distribution X over $\{0,1\}^{n \times \ell}$ is a distribution that is somewhere-$\mathcal{P}$ where $\mathcal{P}$ contains only the uniform distribution on n-bit strings. We use the following properties of somewhere-uniform distributions.

Lemma 4.8 (Properties of somewhere-uniform sources)**.** *Let X be a somewhere-uniform distribution over $\{0,1\}^{n \times \ell}$ then*

- $H^\infty(X) \geq n - \log \ell$.
- *For every $n' < n$ let X' be a distribution over $\{0,1\}^{n' \times \ell}$ obtained by truncating each block of X to length n'. Then, X' is somewhere-uniform.*

Proof. For the first item note that $H^\infty(X, I) \geq H^\infty(X_I)$ and by Lemma 3.2 $H^\infty(X) \geq H^\infty(X, I) - \log \ell$. The second item follows as truncating a uniform distribution gives a uniform distribution. □

Note that by Lemma 4.7 we also have that being close to somewhere-uniform and being somewhere-(close to uniform) is the same.

Remark 4.9 (Ta-Shma's notion of Somewhere-uniform sources)**.** We remark that a slightly different notion of Somewhere-uniform sources was previously considered by Ta-Shma [TS96]. This notion requires that for every value i of the selector I, the distribution $(X_I|I = i)$ is uniform. To make the distinction between the two definitions more clear consider the distribution X over $\{0,1\}^{1\times 2}$ which assigns probability one to the string $(x_1, x_2) = (0, 1)$. Consider the selector I over $\{1, 2\}$ which gives equal probability to the two values. The distribution X_I is uniform and using our jargon this gives that X is a somewhere uniform distribution. However, using Ta-Shma's definition a distribution that assigns probability one to a single element cannot be somewhere uniform.

We use the weaker notion defined in this paper as it suffices for our purposes and is easier to work with. Nevertheless, we remark that our techniques can also be used to give results which use the stronger notion.

5 A Somewhere Condenser

In this section we construct the following object which we call a *somewhere condenser*.

Definition 5.1 (Somewhere Condensers)**.** A function $\mathtt{con} : \{0,1\}^n \to \{0,1\}^{m\times\ell}$ is a *somewhere-condenser* with *entropy requirement* k*, output entropy* k' *and error* ϵ if for every n-bit source X such that $H^\infty(X) \geq k$, $\mathtt{con}(X)$ is ϵ-close to somewhere-$\mathcal{P}$ where $\mathcal{P}$ is the property of having min-entropy at least k'.

The main result of this section is a construction of a somewhere condenser which given a δ-source outputs a constant number of blocks and has output entropy rate 0.9.

Theorem 5.2 (Somewhere condenser)**.** *For every constant* $\delta > 0$*, there exist constants* $\beta, \eta > 0$*,* $\ell \geq 1$ *and a polynomial time computable function* $\mathtt{con} : \{0,1\}^n \to \{0,1\}^{m\times\ell}$ *such that for every sufficiently large* n*,* $\mathtt{con}$ *is a somewhere-condenser with* $m = \beta n$*, entropy requirement* δn*, output entropy* $0.9m$ *and error* $2^{-\eta m}$*.*

We remark that the constant 0.9 can be replaced with any constant smaller than 1. For our application it is only important that this constant is larger than 1/2. The remainder of this section is devoted to proving Theorem 5.2.

5.1 A basic condenser

We start by constructing a "basic condenser" which improves the rate of the output by a small amount. Loosely speaking, sources with rate δ are "condensed" into sources with rate $\delta(1 + \lambda(\delta))$ where λ is an increasing function of δ.[9] The final condenser $\mathtt{con}$ is then obtained by iteratively applying the basic condenser.

[9] A preliminary version of this paper included a different construction of a "basic condenser" see discussion in Remark 5.4.

Theorem 5.3 (Basic condenser)**.** *There exist universal constants $\eta, \alpha > 0$ and $c \geq 1$ such that for every constant $0 < \delta \leq 0.9$, there exist constants $\beta' > 0$, $\ell' \geq 1$ and a polynomial time computable function* $\texttt{bcon} : \{0,1\}^n \to \{0,1\}^{m \times \ell'}$ *such that for every sufficiently large n,* $\texttt{bcon}$ *is a somewhere-condenser with $m = \beta' n$, entropy requirement δn, output entropy $((1+\lambda) \cdot \delta) \cdot m$ for $\lambda = \eta \delta^c$, and error $\epsilon = 2^{-\alpha \delta m}$.*

Proof. When given $\delta > 0$ we use Theorem 3.6 to get an extractor $\texttt{ext}$ for sources of entropy rate $\delta/2$. In this proof it is more convenient to denote the length of the source by m. That is, by Theorem 3.6 there exists $\ell = (1/\delta)^{O(1)}$, $\gamma > 0$ and an ℓ-source extractor $\texttt{ext} : \{0,1\}^{m \times \ell} \to \{0,1\}^m$ with entropy requirement $\delta m/2$ and error $2^{-\gamma m}$. We can assume w.l.o.g. that $\gamma \leq 1/100$. We define $\lambda = \gamma/(3\ell)$, $m = n/\ell$, $\ell' = \ell + 1$ and $\beta' = 1/\ell$. Given a string x of length n we partition it into ℓ blocks of length m which we denote by $x_1, \ldots, x_\ell$ and define:

$$\texttt{bcon}(x) = (x_1, \ldots, x_\ell, \texttt{ext}(x_1, \ldots, x_\ell))$$

Note that the output of $\texttt{bcon}$ indeed consists of $\ell' = \ell + 1$ blocks of length $m = \beta' n$. (We can assume w.l.o.g. that ℓ divides m as for large enough n we can pad an n-bit source with zeros to make it a source of length that is a multiple of ℓ while only slightly decreasing the entropy rate δ).

Let $k' = (1 + \lambda/2) \cdot \delta m$ and consider the property $\mathcal{P}$ of m-bit sources having min-entropy at least k'. Given an n-bit source X with $H^\infty(X) \geq \delta n$ we need to show that $\texttt{bcon}(X)$ is ϵ-close to somewhere-$\mathcal{P}$. We can assume w.l.o.g. that X is a flat source and is uniformly distributed over a set $S \subseteq \{0,1\}^n$. This is because every source X with $H^\infty(X) \geq \delta n$ is a convex combination of flat sources and furthermore, the class of distributions that are ϵ-close to somewhere-$\mathcal{P}$ are closed under convex combinations.

For every $1 \leq i \leq \ell + 1$ we define $H_i = \{y \in \{0,1\}^m : \Pr[\texttt{bcon}(X)_i = y] \geq 2^{-(1+\lambda)\delta m}\}$, namely the "heavy elements" of the distribution $\texttt{bcon}(X)_i$. It follows that for every i, $|H_i| \leq 2^{(1+\lambda)\delta m}$. Let $\epsilon = 2^{-\alpha \delta m}$ where $\alpha = \gamma/3$. We distinguish between two cases:

Case 1: $\Pr[\exists i : \texttt{bcon}(X)_i \notin H_i] \geq 1 - \epsilon$. Let S' be the set of $x \in S$ for which the event above holds, namely:

$$S' = \{x \in S : \exists i,\ \texttt{bcon}(x)_i \notin H_i\}$$

Let X' be the uniform distribution over S'. Note that X and X' are ϵ-close. It is therefore sufficient to show that $\texttt{bcon}(X')$ is somewhere-$\mathcal{P}$. For $x \in S'$ we define $I(x)$ to be the smallest i such that $\texttt{bcon}(x)_i \notin H_i$. We now consider the probability space that consists of X' and $I = I(X')$ and show that indeed $H^\infty(\texttt{bcon}(X')_I) \geq (1 + \lambda/2)\delta m$ for large enough n.

For this purpose we fix some $y \in \{0,1\}^m$ and estimate $\Pr[\texttt{bcon}(X')_I = y]$. In the calculation below we use the fact that for every i such that $y \in H_i$ and every $x \in S'$ such that $I(x) = i$ we have that $\texttt{bcon}(x)_i \neq y$. In addition, we also use the fact that for every i, y,

$$\Pr[\texttt{bcon}(X')_i = y] = \Pr[\texttt{bcon}(X)_i = y | X \in S'] \leq 2 \Pr[\texttt{bcon}(X)_i = y]$$

where the inequality follows using $\Pr[A|B] \leq \Pr[A]/\Pr[B]$ and for large enough n, $\Pr[X \in S'] \geq 1 - \epsilon \geq 1/2$. We now proceed with the calculation.

$$
\begin{aligned}
\Pr[\mathtt{bcon}(X')_I = y] &= \sum_{1\le i\le \ell+1} \Pr[\mathtt{bcon}(X')_i = y \wedge I = i] \\
&= \sum_{i: y\notin H_i} \Pr[\mathtt{bcon}(X')_i = y \wedge I = i] \\
&\le \sum_{i: y\notin H_i} \Pr[\mathtt{bcon}(X')_i = y] \\
&\le \sum_{i: y\notin H_i} 2\Pr[\mathtt{bcon}(X)_i = y] \\
&\le 2(\ell+1)\cdot 2^{-(1+\lambda)\delta m} \\
&\le 2^{-(1+\lambda)\delta m + 2\log \ell} \\
&\le 2^{-(1+\lambda/2)\delta m}
\end{aligned}
$$

Case 2: $\Pr[\forall i : \mathtt{bcon}(X)_i \in H_i] \ge \epsilon$. We are going to show that this is a contradiction and together with the analysis we made in Case 1 this completes the proof. Let S' be the set of all $x \in S$ for which the event above holds, namely:

$$S' = \{x \in S : \forall i,\ \mathtt{bcon}(X)_i \in H_i\}$$

We have that

$$|S'| \ge \epsilon \cdot |S| \ge \epsilon \cdot 2^{\delta n} = \epsilon \cdot 2^{\delta \ell m} = 2^{(\ell-\alpha)\delta m}$$

Note that S' is a subset of $\Pi_{1\le i\le \ell} H_i$. In particular,

$$|H_1| \cdot \ldots \cdot |H_\ell| \ge |S'| \ge 2^{(\ell-\alpha)\delta m}$$

However, as we have that for every $1 \le i \le m$, $|H_i| \le 2^{(1+\lambda)\delta m}$ we can conclude that

$$|H_1| \ge \frac{|S'|}{|H_2| \cdot \ldots \cdot \ldots \cdot |H_\ell|} \ge \frac{2^{(\ell-\alpha)\delta m}}{(2^{(1+\lambda)\delta m})^{\ell-1}} \ge 2^{\delta m(1-\alpha-(\ell-1)\lambda)} \ge 2^{-\delta m/2}$$

where the last inequality follows because $\alpha = \gamma/3$, $\lambda = \gamma/3\ell$ and $\gamma \le 1/100$. The same reasoning gives that for every $1 \le i \le \ell$, $|H_i| \ge 2^{\delta m/2}$. We now consider a probability space with ℓ independent random variables $Y_1, \ldots, Y_\ell$ where each Y_i is an m-bit source that is uniformly distributed over H_i. We are going to get a contradiction by estimating $\Pr[\mathtt{ext}(Y_1, \ldots, Y_m) \in H_{\ell+1}]$ in two ways. We have that for every i, $H^\infty(Y_i) \ge \delta m/2$ and therefore we have that $\mathtt{ext}(Y_1, \ldots, Y_m)$ is $2^{-\gamma m}$-close to uniform. In particular,

$$\Pr[\mathtt{ext}(Y_1, \ldots, Y_m) \in H_{\ell+1}] \le \frac{|H_{\ell+1}|}{2^m} + 2^{-\gamma m} \le \frac{2^{(1+\lambda)\delta m}}{2^m} + 2^{-\gamma m} = 2^{-m(1-\delta-\lambda\delta)} + 2^{-\gamma m} \le 2^{-\gamma m+1}$$

where the last inequality follows because $\delta \le 0.9$, $\lambda = \gamma/3\ell$ and $\gamma \le 1/100$. On the other hand, for every $x \in S'$ we have that $\mathtt{ext}(x_1, \ldots, x_\ell) \in H_{\ell+1}$. Therefore,

$$\Pr[\mathtt{ext}(Y_1, \ldots, Y_\ell) \in H_{\ell+1}] \ge \Pr[(Y_1, \ldots, Y_\ell) \in S'] = \frac{|S'|}{|H_1| \cdot \ldots \cdot |H_\ell|}$$

$$\geq \frac{2^{(\ell-\alpha)\delta m}}{2^{(1+\lambda)\delta m \ell}} = 2^{-\delta m(\alpha+\lambda\ell)} \geq 2^{-2\gamma m/3}$$

where the last inequality follows because $\delta \leq 1$, $\alpha = \gamma/3$ and $\lambda = \gamma/3\ell$. Thus, we obtain a contradiction as $2^{-2\gamma m/3} > 2^{-\gamma m+1}$. □

Remark 5.4 (A basic condenser with four output blocks)**.** We remark that the constants in Theorem 5.3 can be improved and some of them can be made universal constants that do not depend on δ. More specifically, assume that $n = 3p$ where p is a prime (this is w.l.o.g. by padding the input if necessary). Define the function $\texttt{bcon} : \{0,1\}^n \to \{0,1\}^{p\times 4}$ as follows: When given an input $x \in \{0,1\}^n$ split it into x_1, x_2, x_3 where each of these is a string of length p. We can identify such strings with elements in the field with 2^p elements and define:

$$\texttt{bcon}(x) = (x_1, x_2, x_3, f(x_1, x_2, x_3))$$

where

$$f(x_1, x_2, x_3) = x_1 + x_2 \cdot x_3$$

It can be shown that there exists a universal constant $\lambda > 0$ such that for every constant $0 < \delta \leq 0.9$ $\texttt{bcon}$ is a somewhere condenser with entropy requirement δn, output entropy $(1+\lambda)\delta \cdot p$ and error $2^{-\lambda\delta^2}$. Note that this improves Theorem 5.3 in that both the number of blocks and the "expansion factor" λ are now universal constants that do not depend on δ.

The high level idea of the proof can be described as follows: The first step is to show that for three independent δ-sources Y_1, Y_2, Y_3 over $\{0,1\}^p$ the random variable $f(Y_1, Y_2, Y_3)$ is close to a $(1+\lambda)\delta$ source. The second step is to use the argument in the proof of Theorem 5.3 using f instead of the extractor $\texttt{ext}$. (This argument appeared in the conference version of this paper).

Barak, Impagliazzo and Wigderson [BIW04] showed how to implement the first step above for the special case of prime fields (and guaranteeing an "expansion factor" $\lambda = \Omega(\delta)$). The proof of [BIW04] relies on a "sum-product theorem" by Bourgain, Katz and Tao [BKT04]. Using recent improved "sum-product theorems" [TV06, Thm 2.55],[KS09] one can extend the argument of [BIW04] and implement the first step above for fields with 2^p elements using a universal constant $\lambda > 0$. The reader is referred to [TV06] for a survey on "sum-product theorems".

Proving the extension of the results of [BIW04] to fields of size 2^p is not within the scope of this paper and therefore in Theorem 5.3 we chose to present a somewhere condenser that is based on the extractor of [BIW04] and has worse parameters. We remark that the argument presented in this version resembles the one that was independently obtained by Raz [Raz05].

We note that using the improved basic condenser in our constructions of 3-source extractor and 2-source disperser does not affect the parameters in the final results (except for improving the dependance of certain constants on the constant δ).

5.2 Composing condensers: Proof of Theorem 5.2

We now iteratively apply the basic condenser to construct the condenser of Theorem 5.2. We need the following straightforward lemma on composition of somewhere-condensers.

Lemma 5.5 (Composing condensers)**.** *Let $C_1 : \{0,1\}^{n_1} \to \{0,1\}^{\ell_1 \times n_2}$ be a somewhere condenser with entropy requirement k_1, output entropy k_2 and error ϵ_1. Let $C_2 : \{0,1\}^{n_2} \to \{0,1\}^{\ell_2 \times m}$ be a somewhere condenser with entropy requirement k_2, output entropy k' and error ϵ_2. We define $C : \{0,1\}^{n_1} \to \{0,1\}^{\ell_1 \cdot \ell_2 \times m}$ as follows: Identify an index $i \in [\ell_1 \cdot \ell_2]$ as a pair $(i_1, i_2) \in [\ell_1] \times [\ell_2]$ and let*

$$C(x)_{(i_1,i_2)} = C_2(C_1(x)_{i_1})_{i_2}$$

Then C is a somewhere condenser with entropy requirement k_1, entropy output k' and error $\epsilon_1+\epsilon_2$.

Proof. Let X be an n_1-bit source with $H^\infty(X) \geq k_1$. Let $\mathcal{P}^1$ be the property of having min-entropy k_2 and let $\mathcal{P}^2$ be the property of having min-entropy k'. We have that $C_1(X)$ is ϵ_1-close to somewhere-$\mathcal{P}^1$. By Lemma 4.7, $C_1(X)$ is somewhere-$\mathcal{P}^1_{\epsilon_1}$, that is there exists a selector I_1 such that $C_1(X)_{I_1}$ is ϵ_1-close to a distribution Y with $H^\infty(Y) \geq k_2$. We have that $C_2(Y)$ is ϵ_2-close to somewhere-$\mathcal{P}^2$. Since X_{I_1} and Y are ϵ_1-close it follows that $C_2(Y)$ and $C_2(X_{I_1})$ are ϵ_1-close and we can conclude that $C_2(X_{I_1})$ is $(\epsilon_1+\epsilon_2)$-close to somewhere-$\mathcal{P}^2$. That is, there exist a selector I_2 such that $C(X)_{(I_1,I_2)} = C_2(X_{I_1})_{I_2}$ is $(\epsilon_1+\epsilon_2)$-close to somewhere-$\mathcal{P}^2$ as required. □

Theorem 5.2 follows by iteratively applying the basic condenser of Theorem 5.3 and using the composition lemma above. Note that starting with entropy rate $\delta > 0$ each step improves the entropy rate by a factor $(1+\lambda)$. We have that λ is an increasing function of δ and therefore after a constant number of steps (where this constant depends on δ) we obtain rate 0.9.

6 A 2-Source Somewhere-Extractor

In this section, we construct a (2-source) somewhere-extractor, which is a function that given inputs from two independent n-bit δ-sources (where δ is an arbitrarily small constant) outputs a somewhere-uniform distribution.

Definition 6.1 (Somewhere-extractor)**.** We say that

$$\texttt{s_ext} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^{m\times\ell}$$

is a *somewhere-extractor* with *entropy requirement k and error ϵ* if for every two independent n-bit sources X, Y with $H^\infty(X) \geq k$ and $H^\infty(Y) \geq k$ then $\texttt{s_ext}(X,Y)$ is ϵ-close to a somewhere-uniform distribution.

A fixed $y \in \{0,1\}^n$ is *good* for first source X if $\texttt{s_ext}(X,y)$ is ϵ-close to a somewhere uniform distribution. Similarly, a fixed $x \in \{0,1\}^n$ is good for second source Y if $\texttt{s_ext}(x,Y)$ is ϵ-close to a somewhere uniform distribution. (If we omit mentioning for which source a fixed string is good then we mean the first source). We say that $\texttt{s_ext}$ is a *strong* somewhere-extractor with the same parameters as above if for every X, Y as above,

$$\Pr_{y\leftarrow Y}[y \text{ is good for first source } X] > 1-\epsilon$$

and

$$\Pr_{x\leftarrow X}[x \text{ is good for second source } Y] > 1-\epsilon.$$

The main result of this section is a construction of a somewhere-extractor which takes two independent δ-sources and produces a somewhere-uniform distribution with a constant number of blocks.

Theorem 6.2 (Somewhere-extractor)**.** *For every constant $\delta > 0$ there are constants $\ell \geq 1$, $\beta, \eta > 0$ such that for every $m \leq \beta n$ there is a polynomial-time computable function $\texttt{s_ext} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^{m\times\ell}$ such that for every sufficiently large n, $\texttt{s_ext}$ is a strong somewhere-extractor with entropy requirement δn and error $\epsilon \leq 2^{-\eta n}$.*

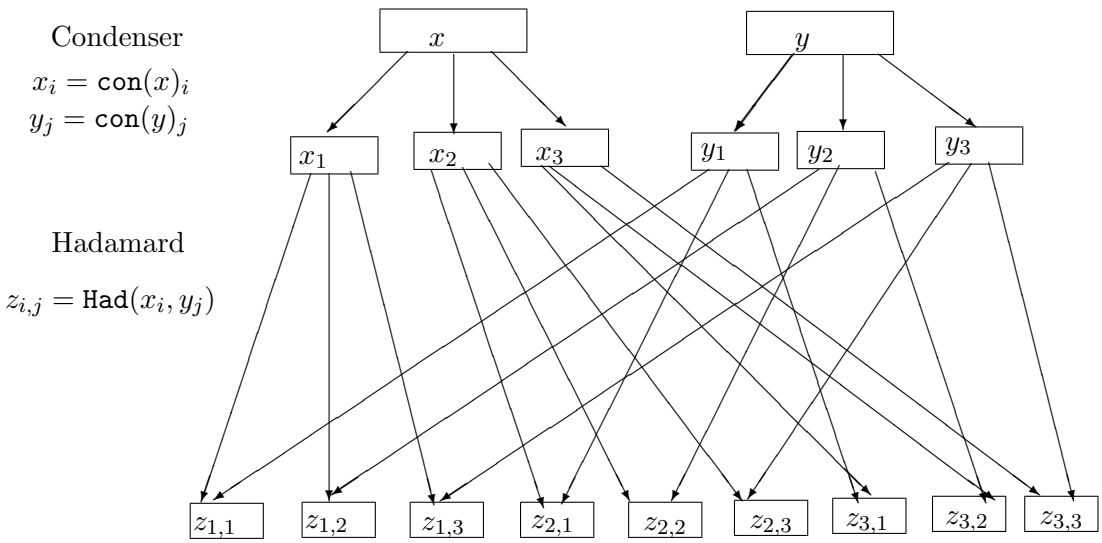

Figure 4: A (2-source) Somewhere-Extractor

6.1 The construction

The outline of our construction is sketched in Figure 4. Roughly speaking, the idea behind the construction is to use the somewhere condenser derived from Theorem 5.2 to obtain from each source a list of blocks one of whom has entropy rate larger than $\frac{1}{2}$. We then apply to every possible pair of blocks the previously known construction of a two-source extractor that works for sources with rate larger than $\frac{1}{2}$ (i.e., for every choice of a block from the first source and a block from the second source we run the two-source extractor on both blocks and output the results). The number of blocks in the output is the product of the number of output blocks of two condensers. Another tool which we need, in addition to Theorem 5.2, is the following result on two-source extractors.

Theorem 6.3 (Two-source extractors [CG88, Vaz87, DEOR04])**.** *There is a polynomial-time 2-source extractor* $\mathtt{Had} : \{0,1\}^{n\times 2} \to \{0,1\}^k$ *with* $k = \Omega(n)$*, entropy requirement* $0.6n$ *and error* $2^{-\Omega(n)}$*. Furthermore, this 2-source extractor is* strong*. More formally, it is a strong somewhere extractor (with* $\ell = 1$ *output blocks) for entropy requirement* $0.6n$ *and error* $2^{-\Omega(n)}$*.*

We remark that the strongness property of the extractor above was first observed in [DEOR04].

Proof of Theorem 6.2. Given δ let β and ℓ be the constants from Theorem 5.2 and let $\mathtt{con}$ be the somewhere condenser from the theorem. We have that $\mathtt{con}$ has entropy requirement δn and error $\epsilon = 2^{-\Omega(\beta n)}$ and outputs ℓ blocks of length $n' \stackrel{def}{=} \beta n$. Let $\mathtt{Had}_{n'}$ be the two source extractor of Theorem 6.3 with input length n'. Our somewhere random extractor will have ℓ^2 blocks. We index such a block by a pair of elements in $[\ell]$ and define

$$\mathtt{s_ext}(x_1, x_2)_{(j_1,j_2)} = \mathtt{Had}_{n'}(\mathtt{con}(x_1)_{j_1}, \mathtt{con}(x_2)_{j_2})$$

By definition, the length of every output block is $\Omega(n') = \Omega(n)$. Let X_1, X_2 be independent n-bit sources with $H^\infty(X_1) \geq \delta n$ and $H^\infty(X_2) \geq \delta n$. For $i \in \{1,2\}$, by definition of $\mathtt{con}_i$, there exist random variable I_i such that $\mathtt{con}_i(X_i)_{I_i}$ is ϵ_i close to have rate $0.9 > 0.6$. This implies

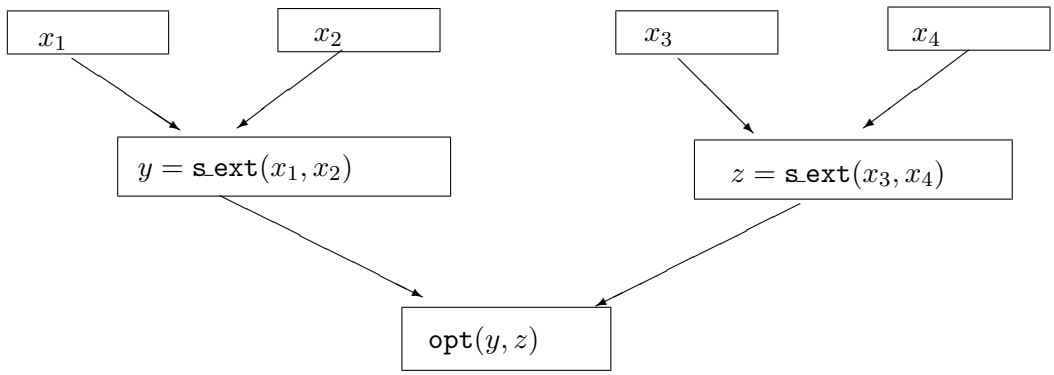

Figure 5: A 4-source extractor

that $\texttt{s_ext}(X_1, X_2)_{(I_1,I_2)}$ is ϵ-close to uniform distribution for $\epsilon = \left(\epsilon_1 + \epsilon_2 + 2^{-\Omega(n')}\right) = 2^{-\Omega(n')}$. Using Lemma 4.7 we conclude that the output of the extractor is ϵ-close to a somewhere uniform distribution. Moreover, using the fact that `Had` is a strong extractor it follows that `s_ext` is also strong. Since $n' = \Omega(n)$, we conclude there exist a constant $\eta > 0$ (that depends on δ) such that $\epsilon \leq 2^{-\eta n}$. Finally by Lemma 4.8 we can truncate the output blocks to length m if we want to reduce the block length to $m \leq \beta n$.

7 A 3-Source Extractor

In this section we prove Theorem 1.1. That is we construct a three source extractor for sources with rate δ for any $\delta > 0$. This is the first construction of an ℓ-source extractor that works for sources with rate δ such that the constant $\ell \cdot \delta$ can be made arbitrarily small.

The two tools we use in this construction is the somewhere random extractor of Section 6 and an *optimal* two source extractor with very small input size. For very small input sizes it is sufficient to show the existence of a 2-source extractor using the probabilistic method as one can find such an extractor by brute force. The next Lemma describes the extractor we use in our construction.

Lemma 7.1 (Optimal Extractor)**.** *For every integers d and $m = \lfloor \log d \rfloor$, there exists a $2^{5d^{14}}$-time computable 2-source extractor* $\texttt{opt} : \{0,1\}^{d \times 2} \to \{0,1\}^m$ *with $6 \log d$-entropy requirement and distance $1/d$.*

The proof of Lemma 7.1 follows by a standard calculation and is given for completeness in Appendix A.

7.1 An Appetizer: A 4-Source Extractor

Before proving Theorem 1.1 we prove a slightly relaxed version of it, which demonstrates the main idea behind the proof. That is, we prove that there exists a 4-source extractor with the same parameters. The construction of this extractor is quite simple.

The construction Given the constants δ, ϵ and m we choose d large enough to satisfy:

- $\frac{1}{d} < \frac{\epsilon}{2}$ (recall that ϵ is the desired statistical distance)

- $\log d \geq m$
- $d/\ell - \log \ell \geq 6 \log d$. (Here ℓ is the constant from Theorem 6.2 that is defined as a function of δ).

Next apply Theorem 6.2 to get a somewhere random extractor with entropy requirement δn. Let ℓ, β and η be the constants whose existence is guaranteed by the theorem and note that since n is large $\beta n > d/\ell$. We can truncate the length of blocks to d/ℓ. We denote this somewhere-extractor by $\texttt{s_ext}_{d/\ell}$. This extractor outputs ℓ blocks of length d/ℓ so overall it outputs d bits. Its output is $2^{-\eta n}$-close to somewhere-uniform.

Let $\texttt{opt}$ be the "optimal extractor" from Lemma 7.1 with input length d. As d is a constant it follows that $\texttt{opt}$ can be computed in constant time. Note that the output length of $\texttt{opt}$ is $\log d \geq m$, so we can truncate it to length m. Our 4-sample extractor $\texttt{ext}_4 : \{0,1\}^{n \times 4} \to \{0,1\}^m$ is defined as follows (see also Figure 5):

$$\texttt{4ext}(x_1, x_2, x_3, x_4) = \texttt{opt}\Big(\texttt{s_ext}_{d/\ell}(x_1, x_2)\ ,\ \texttt{s_ext}_{d/\ell}(x_3, x_4)\Big)$$

We now observe that this indeed gives a 4-source extractor with the required parameters.

Theorem 7.2 (4-source extractor)**.** *For every constants $\delta, \epsilon > 0$ and $m \in \mathbb{N}$, and for every sufficiently large integer n there exists a* poly(n)*-time computable 4-source extractor* $\texttt{4ext} : \{0,1\}^{n \times 4} \to \{0,1\}^m$ *with entropy requirement δn and error ϵ.*

Proof. Let $X_1, \ldots, X_4$ be independent distributions over $\{0,1\}^n$ all with entropy rate at least δ. We have that $\texttt{s_ext}(X_1, X_2)$ is $2^{-\eta n}$-close to a somewhere random distribution with ℓ blocks. Recall that a somewhere random distribution with ℓ blocks of length d/ℓ has min-entropy at least $d/\ell - \log \ell > 6 \log d$. It follows that $\texttt{s_ext}(X_1, X_2)$ is $2^{-\eta n}$-close to having min-entropy $6 \log d$ and the same holds for $\texttt{s_ext}(X_3, X_4)$. Thus, we apply $\texttt{opt}$ with two independent distributions which meet its entropy requirement and can conclude that the output is $(2 \cdot 2^{-\eta n} + 1/d)$-close to uniform. We have that $1/d < \epsilon/2$ and therefore for large enough n, $2 \cdot 2^{-\eta n} + 1/d \leq \epsilon$, as required. □

7.2 Down from 4 Sources to 3

To construct a 3-source extractors we use the fact that our somewhere random extractor is in fact *strong*. Using this property we are able to show that we can "reuse" one of the sources without harming the input distribution.

The construction: Given the constants δ, ϵ and m we choose the parameters as in the construction for 4-sources. Our 3-source extractor is given by:

$$\texttt{3ext}(x_1, x_2, x_3) = \texttt{opt}\Big(\texttt{s_ext}_{d/\ell}(x_1, x_2)\ ,\ \texttt{s_ext}_{d/\ell}(x_3, x_2)\Big)$$

Proof of Theorem 1.1. Let X_1, X_2, X_3 be independent distributions over $\{0,1\}^n$ with entropy rate at least δ. Recall that $\texttt{s_ext}$ is a strong somewhere extractor, and recall the notion of a "good" input from Definition 6.1. For $i \in \{1,3\}$, let $B_i = \{x : x \text{ is bad for } X_i\}$. We have that $\Pr_{x \leftarrow X_2}[x \in B_i] \leq 2^{-\eta n}$. Let $B = B_1 \cup B_3$. It follows that if we fix random value of x_2 then with probability $1 - 2 \cdot 2^{-\eta n}$ we obtain that $x_2 \notin B$. For every such fixed $x_2 \notin B$, the distributions $\texttt{s_ext}(X_1, x_2)$ and $\texttt{s_ext}(X_3, x_2)$ are independent. Furthermore, each one of them is $2^{-\eta n}$-close to a somewhere random distribution with ℓ blocks of length d/ℓ. As in the case of 4 sources, such a distribution meets the entropy requirements of $\texttt{opt}$ and therefore for all $x_2 \notin B$, $\texttt{opt}\big(\texttt{s_ext}(X_1, x_2), \texttt{s_ext}(X_3, x_2)\big)$ is

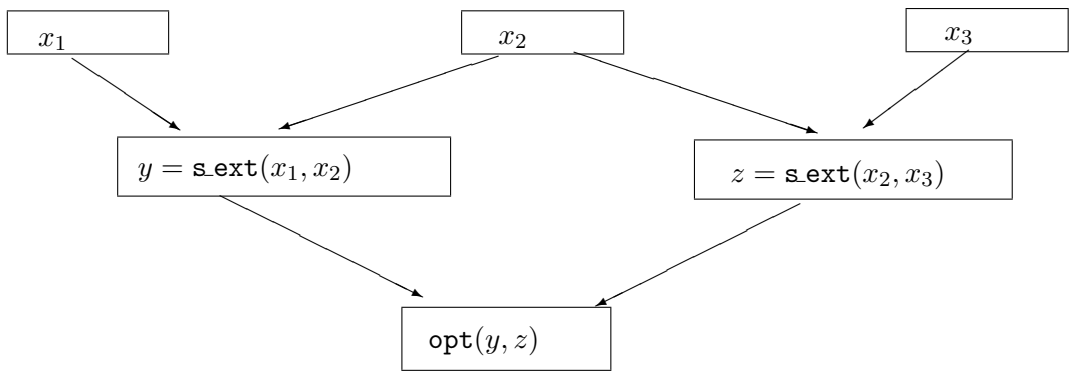

Figure 6: A 3-source extractor

$(2 \cdot 2^{-\eta n} + 1/d)$-close to uniform distribution. This implies that $\texttt{opt}\big(\texttt{s_ext}(X_1, X_2), \texttt{s_ext}(X_3, X_2)\big)$ is $(4 \cdot 2^{-\eta n} + 1/d)$-close to uniform. Once again, we note that $(4 \cdot 2^{-\eta n} + 1/d) \leq \epsilon$ for large enough n. □

8 A 2-Source Disperser

In this section we construct a 2-source disperser for sources of any constant rate $\delta > 0$. This proves Theorem 1.2. We first restate Theorem 1.2 using the notation of Section 3.

Theorem 8.1 (2-source disperser)**.** *For every two constants $\delta > 0$ and $m \in \mathbb{N}$ there exists a polynomial time computable function* $\texttt{disp} : \{0,1\}^{n \times 2} \to \{0,1\}^m$ *such that for every sufficiently large n,* $\texttt{disp}$ *is a 2-source disperser with entropy requirement δn and zero error.*

As mentioned in the introduction our techniques give an object that is stronger than a disperser (although weaker than an extractor). A disperser guarantees that for any two independent δ-sources X and Y and for any $z \in \{0,1\}^m$, $\Pr[Disp(X,Y) = z] > 0$. However, dispersers are allowed to assign very small probabilities to some values z. Our construction has the following stronger property:

Theorem 8.2 (stronger notion of 2-source disperser)**.** *For every two constants $\delta > 0$ and $m \in \mathbb{N}$ there exists a polynomial-time computable function* $\texttt{disp} : \{0,1\}^{n \times 2} \to \{0,1\}^m$ *and a constant $\vartheta = \vartheta(m, \delta) > 0$, such that for every sufficiently large n, and for every two independent n-bit δ-sources X, Y and for every $z \in \{0,1\}^m$ we have*

$$\Pr\left[\texttt{disp}(X,Y) = z\right] \geq \vartheta.$$

Theorems 8.1,8.2 give new explicit constructions of bipartite Ramsey graphs as stated in Corollary 1.3.

Proof. (of Corollary 1.3) To prove this corollary, let $V_1 = V_2 = \{0,1\}^n$ be two sets of vertices of size $N = 2^n$. Define a 2-coloring of the edges of the complete bipartite graph $K_{N,N}$ with parts V_1 and V_2 as follows. The color of an edge $(x, y) : x \in V_1, y \in V_2$ is $\texttt{disp}(x, y)$. Then the above theorem implies that for every two subsets $A \subseteq V_1$ and $B \subseteq V_2$ of size at least $N^\delta = 2^{\delta n}$ and for every color, the constant proportion of edges between A and B has this color. In particular, this coloring has no monochromatic induced subgraph of size N^δ by N^δ. □

8.1 The Construction

Our goal is to construct a disperser for two independent sources. More precisely, we are given parameters $\delta > 0$ and an integer m. We think of these parameters as constants and for any sufficiently large n (as a function of δ and m) we construct a function $\texttt{disp} : \{0,1\}^{n\times 2} \to \{0,1\}^m$ which is a 2-source disperser with entropy requirement δn.

8.1.1 Partitions

Our construction considers various partitions of the input strings $x, y \in \{0,1\}^n$. We divide $[n]$ into $t \stackrel{def}{=} 10/\delta$ equal blocks of length $\delta n/10$. (We assume from now on that t divides n. This is without loss of generality as we can always add dummy bits to inputs to make the requirement hold, at the cost of only slightly decreasing δ.) For each $i \in [t]$ we consider a partition of $[n]$ into three segments in the following way. The first segment S_1 consists of the blocks $1, \ldots, i-1$. The second segment S_2 consists of the block i and the third segment S_3 consists of the blocks $i+1, \ldots, t$. More formally, let $q = \delta n/10$ be the length of blocks then $S_1 = \{1, \ldots, (i-1)q\}$, $S_2 = \{(i-1)q+1, \ldots, iq\}$ and $S_3 = \{iq+1, \ldots, n\}$. (It is possible that some segments are empty.) For a string $x \in \{0,1\}^n$, $i \in [t]$ and $j \in [3]$ we define x^i_j to be the restriction of x to the indices in the j'th segment. More precisely, $x^i_j \stackrel{def}{=} x_{S_j}$ where S_j is the j'th segment that is defined using the block i. We often omit i if it is clear from the context and then x_1, x_2, x_3 refer to the substrings x^i_1, x^i_2, x^i_3.

A *partition* I is a pair (i_1, i_2) that is used to partition two strings x, y of length n. More precisely, given a partition I and strings $(x, y) \in \{0,1\}^{n\times 2}$ we define $x^I_j \stackrel{def}{=} x^{i_1}_j$ and $y^I_j = y^{i_2}_j$. That is, i_1 is used to partition x and i_2 is used to partition y. We often omit I if it is clear from the context (that is for a fixed partition I we may write expressions like x_2 or y_3). Note that the number of different partitions is $t^2 = 100/\delta^2$ which is a constant that depends only on δ.

We also define a partial order $\preceq$ on partitions in the following way: Given two partitions $I = (i_1, i_2)$ and $I' = (i'_1, i'_2)$ we say that $I \preceq I'$ if $i_1 \le i'_1$ *and* $i_2 \le i'_2$. Note that two partitions may be incomparable according to this partial order. We say that $I \not\preceq I'$ if it does not hold that $I \preceq I'$ (in particular, I and I' are incomparable then $I \not\preceq I'$).

8.1.2 Ingredients

The construction of the disperser uses the same ingredients used in the proof of Theorem 1.1 but in a significantly more complicated way. In order to make the presentation modular we review the properties of the objects that we need. (The properties are also summarized in Figure 7).

A somewhere extractor We make use of a strong somewhere extractor of Theorem 6.2. More precisely, we assume that there exist positive constants ℓ, β and η (that may depend on δ) such that we have a polynomial time computable strong somewhere extractor $\texttt{s_ext} : (\{0,1\}^n)^2 \to \{0,1\}^{\beta n\times \ell}$ with entropy requirement $\delta^5 n/1000$ and distance $2^{-\eta n}$. We use the notation $\texttt{s_ext}_b$ to denote the application where the ℓ outputs are truncated to length b. In the formal presentation below we abuse the notation and allow $\texttt{s_ext}$ to receive inputs of length varying between $\delta^3 n$ and n by padding the input with zeros if necessary. Summing up, for every constant b and every two independent distributions X, Y on strings of lengths n_1, n_2 respectively, where n_1, n_2 are between $\delta^3 n$ and n, and assuming $H^\infty(X) \ge \delta^5 n/1000$ and $H^\infty(Y) \ge \delta^5 n/1000$ we have that $\texttt{s_ext}_b(X, Y)$ is $o(1)$-close to a somewhere-uniform distribution.

A non-constructive optimal 2-source extractor for sources of constant size We use Lemma 7.1 to obtain a 2-source extractor $\texttt{opt} : \{0,1\}^{d\times 2} \to \{0,1\}^m$ with a constant d large enough so that $\log d > 100m$ and $\frac{d}{\ell} - \log \ell > 10 \log d$. By the lemma we have that this extractor is computable in polynomial time and has error $1/d \leq 2^{-100m}$ and entropy requirement $6 \log d$. The requirements on the parameters are essentially similar to those made in Section 7.1 and are used to guarantee that $\texttt{opt}$ can be applied on the output of $\texttt{s_ext}_{d/\ell}$.

8.1.3 Definition of the disperser disp

Given input strings x, y of length n we do the following:

1. For every partition I we will define a polynomial time procedure $\texttt{disp}_I(x, y)$. The final output of the disperser will be $\texttt{disp}_I(x, y)$ for some partition I that will be chosen as a function of x and y. We next explain how to choose the partition I.

2. For every partition I we will define a polynomial time procedure $\texttt{test}_I(x, y)$ that outputs a Boolean value. We say that the partition "passes" the test if $\texttt{test}_I(x, y)$ accepts.

3. The operation of the disperser is as follows: When given the inputs x, y we go over all partitions I and compute $\texttt{test}_I(x, y)$. We choose the partition I that is minimal with respect to the partial order $\preceq$ amongst the partitions that passed the test. If the minimal partition is not unique we choose an arbitrary minimal partition. If no partition passes the test we choose an arbitrary partition. Finally, once we decide on a partition I we set $\texttt{disp}(x, y) = \texttt{disp}_I(x, y)$.

The procedure $\texttt{disp}_I(x, y)$: Given a partition I, we define:

$$\texttt{disp}_I(x, y) = \texttt{opt}(\texttt{s_ext}_{d/\ell}(x_1, y_2), \texttt{s_ext}_{d/\ell}(y_1, x_2))$$

The procedure $\texttt{disp}_I$ is described pictorially in Figure 9.

The procedure $\texttt{test}_I(x, y)$. Given a partition I, we define $\texttt{test}_I(x, y)$ as follows:

1. Choose k to be large enough constant so that $\ell^4 t^2/\delta^3 2^{-k} \leq 2^{-10m}$ (note that this is possible as all other parameters participating in the inequality above are constants). Let $c_1(x, y) = \texttt{s_ext}_k(x_3, y)$ and let $c_2(x, y) = \texttt{s_ext}_k(y_3, x)$ (if x_3 is of length zero we set c_1 to be a fixed string of length k and if y_3 is of length zero we set c_2 to be a fixed string of length k). Let $c = c(x, y)$ be the concatenation of $c_1(x, y)$ and $c_2(x, y)$. Note that $|c| = 2\ell k$ since the number of output blocks in all somewhere-extractors which we are using is ℓ. We refer to c as the "challenge string" of x, y for partition I.

2. We now consider another way of partitioning a string x of length n into blocks. We assume (again without loss of generality) that $\delta^3 n$ divides n and partition $[n]$ into $p = 1/\delta^3$ intervals of length $\delta^3 n$. More precisely, for $1 \leq v \leq p$ let $x[v]$ denote the v'th block that is $x[v] = x_{\{(v-1)\delta^3 n+1,\ldots,v\delta^3 n\}}$. For every $(v_1, v_2) \in [p]^2$ let $c_{v_1,v_2}(x, y) = \texttt{s_ext}_{2\ell k}(x[v_1], y[v_2])$. Note that there is at most a constant number of such possible pairs $(v_1, v_2) \in [p]^2$. We refer to $c_{v_1,v_2} = c_{v_1,v_2}(x, y)$ as a "response string".

3. The procedure $\texttt{test}_I(x, y)$ computes $c(x, y)$. It also computes $c_{v_1,v_2}(x, y)$ for all $(v_1, v_2) \in [p]^2$. Note that for every pair (v_1, v_2) the length of each of the ℓ blocks in c_{v_1,v_2} is exactly $2\ell k$ which

is the length of the challenge string c. If there exists a pair (v_1, v_2) such that c is a sub-block of c_{v_1,v_2} then we say that "the challenge is responded" and set $\texttt{test}_I(x, y) = 1$. Otherwise, $\texttt{test}_I(x, y) = 0$.

This concludes the description of the construction. To assist the reader we summarize all the ingredients and parameters in Figure 7 and Figure 8.

Figure 7: Ingredients used in the construction of `disp`

- A somewhere random extractor $\texttt{s_ext}_b$. This extractor can be applied on any two independent sources with lengths varying from $\delta^3 n$ to n and has entropy threshold $\delta^5 n/1000$. It extracts ℓ blocks of length b where b only needs to satisfy that $0 < b \leq \beta n$ for some constant $\beta > 0$. It has error $2^{-\eta n}$ for some constant $\eta > 0$.
- A 2-source extractor `opt` that can extract m bits from any two independent sources of length d and entropy requirement $6 \log d$ where $d > m$ is a constant that is chosen so that the error of `opt` is smaller than 2^{-100m}.

Figure 8: Parameters used in the construction

Name	**Description**	**Notes**
n	Length of the two inputs of `disp`	
$\delta > 0$	Entropy rate of the two input sources	Assumed to be a constant independent of n
m	Required output length of `disp`	Assumed to be a constant independent of n
$t = 10/\delta$	Number of blocks in partitions	
$p = 1/\delta^3$	Number of blocks in defining response strings	We partition any string x of length n into p blocks of length $\delta^3 n$. The v'th block is denoted by $x[v]$
ℓ	Number of output blocks of `s_ext`	A constant that depends only on δ
k	Length of blocks in the challenge string	A constant chosen by the construction so that $\ell^4 t^2/\delta^3 2^k \leq 2^{-10m}$

8.2 An informal overview of the proof

In this section we attempt to give a detailed informal overview of the proof. The reader may skip to the formal proof at any point.

The outline of the proof We show that for every two independent sources X, Y with entropy rate δ there exists a partition $\tilde{I}$ and subsources $\tilde{X}$ of X and $\tilde{Y}$ of Y such that $\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ is (very close to) uniformly distributed. Furthermore, we show that with high probability $\tilde{I}$ is the partition selected when choosing $x \leftarrow \tilde{X}$ and $y \leftarrow \tilde{Y}$. Together, these two properties imply that `disp` is a 2-source disperser as for any two independent δ-sources X, Y and any string z of length m:

$$\Pr[\texttt{disp}(\tilde{X}, \tilde{Y}) = z] \approx \Pr[\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y}) = z] \approx 2^{-m} > 0$$

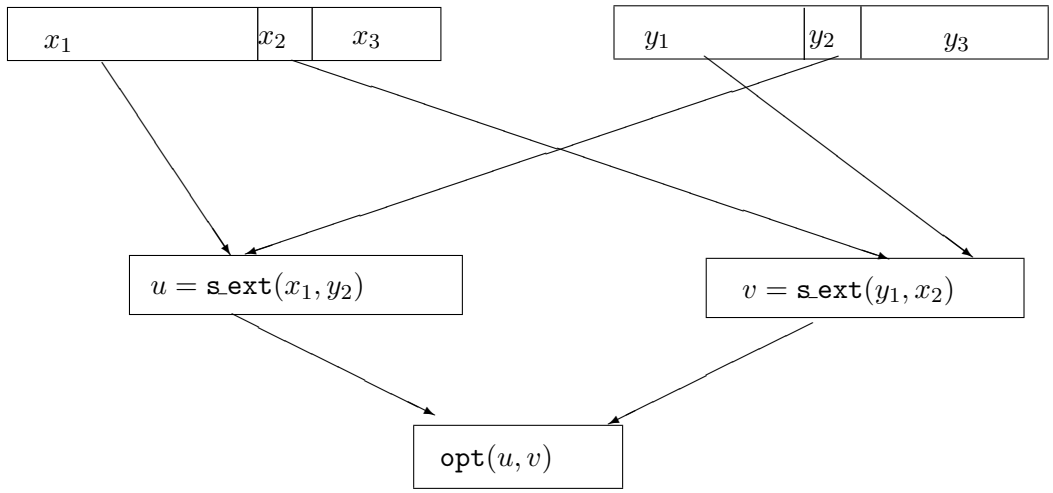

Figure 9: The output $\texttt{disp}_I(x, y)$ with respect to partition I.

Any event that occurs with positive probability under $\tilde{X}, \tilde{Y}$ occurs with positive (though possibly much smaller) probability under X, Y and so:

$$\Pr[\texttt{disp}(X, Y) = z] > 0$$

In fact, we will be able to show that the probability above is greater than some constant $c > 0$ where c depends on δ and m but not on n. We do not explain why this follows in this high level overview.

Nicely structured sources It is not hard to show that for any source X with constant rate there exists a subsource X' which is "nicely structured" in the sense that there is a partition $\tilde{I}$ such that X'_3 is fixed (that is it contains no entropy) while X'_2 has constant rate and X'_1 has constant rate even when conditioned on X'_2. We apply the same reasoning on the second source Y to get an independent subsource Y' with analogous properties. The procedure $\texttt{disp}_I$ is designed so that $\texttt{disp}_{\tilde{I}}$ extracts randomness from every two independent sources that are nicely structured. We remark that the subsources $\tilde{X}, \tilde{Y}$ mentioned above are going to be subsources of X' and Y' which in addition to being nicely structured will have other properties that will insure that the "correct partition" $\tilde{I}$ is selected by the disperser.

The procedure $\texttt{disp}_I$ The procedure $\texttt{disp}_I$ (see also Figure 9) is essentially the 4-source extractor $\texttt{4ext}$ of Section 7.1 when applied on x_1, y_2, y_1, x_2. (A technicality is that these inputs have different lengths). $\texttt{disp}_I$ is designed to work in the special case when applied on independent sources $\tilde{X}, \tilde{Y}$ that are "nicely structured" as explained above. Loosely speaking in such sources the four parts are "sufficiently independent" so that the argument for $\texttt{4ext}$ can be applied.

A little bit more precisely, we consider the case when the variables $\tilde{X}_2, \tilde{Y}_2$ are fixed to some values x_2, y_2. When conditioning on this event (which we denote by E) we have that $\texttt{opt}$ is applied on the distributions $\texttt{s_ext}_{d/\ell}(\tilde{X}_1, y_2)$ and $\texttt{s_ext}_{d/\ell}(\tilde{Y}_1, y_2)$ (conditioned on the event E). The important thing to notice is that these two distributions are independent (as each one depends only on one

of the sources). We would like to say that both these distributions are (close to) somewhere random as in that case the procedure $\texttt{disp}_{\tilde{I}}$ applies $\texttt{opt}$ on two independent distributions of length $\ell \cdot (d/\ell) = d$ which have entropy roughly d/ℓ which means that $\texttt{opt}$ produces an output that is close to uniform. For this purpose we use the strongness of the somewhere extractor $\texttt{s_ext}$: that is that for any constant rate distribution (e.g. $\tilde{X}_1$) most fixings y_2 of $\tilde{Y}_2$ are good in the sense that $\texttt{s_ext}_{d/\ell}(\tilde{X}_1, y_2)$ is (close to) somewhere random. We use this to show that for most fixings x_2, y_2 of $\tilde{X}_2, \tilde{Y}_2$ the two distributions above are indeed independent and (close to) somewhere random.

The actual argument is more technical as in the analysis above the distributions $\texttt{s_ext}_{d/\ell}(\tilde{X}_1, y_2)$ and $\texttt{s_ext}_{d/\ell}(\tilde{Y}_1, y_2)$ are conditioned on the event E. Thus for example,

$$(\texttt{s_ext}_{d/\ell}(\tilde{X}_1, y_2)|E) = (\texttt{s_ext}_{d/\ell}((\tilde{X}_1|\tilde{X}_2 = x_2), y_2)$$

Avoiding technical details in this high level outline we mention that the fact that $\tilde{X}_1$ has constant rate conditioned on $\tilde{X}_2$ plays a crucial role as when conditioning on the event $\{\tilde{X}_2 = x_2\}$ the distribution of the random variable $\tilde{X}_1$ may change but it still retains constant rate. We also need to deal with the fact that while for every x_2 the distribution $(\tilde{X}_1|\tilde{X}_2 = x_2)$ has constant rate and therefore most fixings y_2 of $\tilde{Y}_2$ are good, it could be the case that for different choices of x_2 the set of good y_2's varies. The reader is referred to the proof for the precise details.

The procedure $\texttt{test}_I$: Our goal is to show that there exist two independent nicely structured subsources $\tilde{X}, \tilde{Y}$ of X, Y on which $\texttt{test}_I$ selects the "correct partition" $\tilde{I}$ with high probability. That is we would like the test to pass with probability one on the partition $\tilde{I}$ while it should fail w.h.p. on any partition I that is smaller or incomparable to $\tilde{I}$.

Let $\tilde{X}$ and $\tilde{Y}$ be independent nicely structured sources. We have that when partitioning according to $\tilde{I}$ the third segment $\tilde{X}_3^{\tilde{I}}$ is fixed while the second segment $\tilde{X}_2^{\tilde{I}}$ has constant rate. Consider some partition I that is smaller or incomparable to $\tilde{I}$. Note that this means that in one of the sources (w.l.o.g. the first one) I has a third segment that is longer than that of $\tilde{I}$ and in particular contains the second segment of $\tilde{I}$. It follows that when partitioning according to I, the third segment $\tilde{X}_3^{I}$ has constant rate. Loosely speaking, the procedure $\texttt{test}$ is designed to distinguish between the case where the third segment is fixed (in which case we want it to pass) and the case that the third segment has constant rate (in which case we want it to fail).

More formally, we will show that for any two independent nicely structured sources $\texttt{test}$ indeed fails with high probability when applied with a partition on which the third segment has constant rate. We furthermore show, that for any two independent nicely structured sources X', Y' there exist independent subsources $\tilde{X}$ of X and $\tilde{Y}$ of Y such that $\tilde{X}, \tilde{Y}$ are also nicely structured and furthermore $\texttt{test}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ passes with probability one. Note that this is indeed sufficient for our purposes and gives that $\texttt{disp}$ is a disperser. This is because given independent sources X, Y of rate δ we argued that there exist nicely structured independent subsources X' of X and Y' of Y and these in turn have nicely structured independent subsources $\tilde{X}, \tilde{Y}$ with the properties above. On these subsources the correct partition $\tilde{I}$ is indeed selected with high probability and we apply $\texttt{disp}_{\tilde{I}}$ with the correct partition.

Selecting the correct partition It remains to show that any two independent nicely structured sources have nicely structured independent subsources on which the correct partition is selected with high probability. Let $\tilde{X}, \tilde{Y}$ be independent nicely structured sources. We now explain how the procedure $\texttt{test}$ achieves the goal above. We start by identifying properties of the challenge and response strings.

Properties of the challenge string: The first part of the challenge string is defined by $c_1(x, y) = \texttt{s_ext}_k(x_3, y)$. When partitioning according to an I that is smaller or incomparable to $\tilde{I}$ the third segment $\tilde{X}_3^I$ has constant rate and the two inputs of $\texttt{s_ext}$ meet its entropy requirements. Thus, $c_1(\tilde{X}, \tilde{Y})$ is (close to) somewhere random. A somewhere random distribution with blocks of length k has min-entropy at least $\approx k$ and so the challenge string c (which contains c_1 as a substring) has min-entropy k.

We will need a stronger property, namely that for every choice of v_1, v_2 and response string c_{v_1,v_2} and for every value r the challenge string $c(\tilde{X}, \tilde{Y})$ has min-entropy k even when conditioned on the event $\{c_{v_1,v_2}(\tilde{X}, \tilde{Y}) = r\}$. This follows because response strings depend only on short substrings of $x[v_1]$ of x and $y[v_2]$ of y and therefore $\texttt{s_ext}_k(x_3, y)$ can extract randomness even when these two short substrings are fixed to arbitrary values. This means that the randomness extracted is independent of any response string.

It follows that for any response string c_{v_1,v_2} it is unlikely that $c(\tilde{X}, \tilde{Y})$ appears as a sub-block in the response string $c_{v_1,v_2}(\tilde{X}, \tilde{Y})$. We can control this probability by controlling the constant k. It follows by a union bound over all p^2 response strings that it is very unlikely that the challenge string is responded by *any* response string and therefore $\texttt{test}_I(\tilde{X}, \tilde{Y})$ rejects w.h.p.

On the other hand when partitioning according to $\tilde{I}$ the third segments $\tilde{X}_3^{\tilde{I}}, \tilde{Y}_3^{\tilde{I}}$ are fixed to some strings x_3', y_3'. We would like to say that in that case c is fixed. While this is not true as stated, it is true that in this case $c_1 = \texttt{s_ext}_k(x_3', \tilde{Y})$ is a function only of $\tilde{Y}$. Let c_1' be the most likely value of $c_1(\tilde{X}, \tilde{Y})$ and note that c_1' occurs with probability at least 2^{-k}. We consider the subsource $(\tilde{Y} | \texttt{s_ext}_k(x_3', \tilde{Y}) = c_1')$ of Y and note that this is indeed a subsource of $\tilde{Y}$ as the event on which we condition is determined by Y. The min-entropy of this subsource has reduced by at most k which is a constant that is negligible compared to the entropy present in each of the segments of $\tilde{Y}$. Doing the same process for c_2 we fix c_2 to its most likely value c_2' and obtain a large subsource of $\tilde{X}$. In these new subsources we have that the challenge string is indeed fixed. Note that these two new subsources are nicely structured (as almost no entropy was lost when fixing each if the components of challenge string and so the entropy in each one of the segments did not drop significantly). We can forget about the old sources $\tilde{X}, \tilde{Y}$ and replace them by the new subsources and everything we've said so far still holds for the new subsources as they are also independent and nicely structured. Thus, from now on we refer to the new subsources as $\tilde{X}, \tilde{Y}$. We will now observe that once the challenge string is fixed we can make the response string respond to it.

Properties of the response strings: At this point we are considering the partition $\tilde{I}$ and nicely structured independent sources $\tilde{X}, \tilde{Y}$ such that the challenge string $c(\tilde{X}, \tilde{Y})$ is fixed. As both $\tilde{X}, \tilde{Y}$ have constant rate we can expect that there exists a pair $(v_1, v_2) \in [p]^2$ so that both $\tilde{X}[v_1], \tilde{Y}[v_2]$ have constant rate. (This is true if we were working with Shannon entropy rather than min-entropy as for the first source X we have that $H(X) \leq \sum_{1 \leq v \leq p} H(X[v])$ and we can take the block v_1 such that $H(X[v_1])$ is maximal and this block has rate at least that of the initial source). The case of min-entropy is more difficult but we choose to ignore this technicality in this high level intuition and assume that there exists such a pair (v_1, v_2) with the aforementioned property. For this pair (v_1, v_2) the response string $c_{v_1,v_2}(\tilde{X}, \tilde{Y}) = \texttt{s_ext}_{2\ell k}(\tilde{X}[v_1], \tilde{Y}[v_2])$ is the application of a somewhere extractor on independent constant rate sources. Thus $c_{v_1,v_2}(\tilde{X}, \tilde{Y})$ is somewhere random. Let us imagine that the first output block is random. As the first block is of length $2\ell k$ it follows that the probability that this block is equal to the fixed challenge string is $2^{-2\ell k}$ and in this case the challenge is responded and the test passes. More precisely, let E be the event $\{\texttt{test}_{\tilde{I}}(\tilde{X}, \tilde{Y}) = 1\}$ we have just shown that in the sources $\tilde{X}, \tilde{Y}$, E occurs with positive (but very small) probability.

This is not sufficient for our purposes. We want E to occur with probability that approaches

one. Our approach is to restrict our attention to the distributions $(\tilde{X}|E)$ and $(\tilde{Y}|E)$ and so in these new sources E occurs with probability one. An obvious problem is that these distributions are not necessarily independent as the event E "mixes both sources" (more formally it is not necessarily the case that $(\tilde{X}|E)$ is a subsource of X). At this point we use the fact that c_{v_1,v_2} depends only on short substrings of x and y. Let $\hat{x}, \hat{y}$ be values of $\tilde{X}[v_1], \tilde{X}[v_2]$ such that $\texttt{s_ext}(\hat{x}, \hat{y})$ contains the fixed challenge string c as a substring. Rather than conditioning on E we will condition on the event $E' \subseteq E$ defined by:

$$E' = \{\tilde{X}[v_1] = \hat{x} \text{ and } \tilde{Y}[v_2] = \hat{y}\}$$

Note that conditioning on E' also fixes the response string so that the challenge is responded, however it has the advantage that the distributions $(\tilde{X}|E')$ and $(\tilde{Y}|E')$ are independent (and are in fact subsources of $\tilde{X}, \tilde{Y}$). Furthermore, these two distributions are also nicely structured as fixing the values of $\tilde{X}[v_1], \tilde{Y}[v_2]$ which are relatively short strings cannot lose too much entropy.

We replace the previous sources $\tilde{X}, \tilde{Y}$ with the new subsources and note that these subsources of the initial sources X, Y are nicely structured and in addition have the property that the correct partition $\tilde{I}$ is selected w.h.p. Thus, these subsources have all the properties that we wanted.

To summarize, when $\texttt{disp}$ is applied on independent δ-sources X, Y we have shown that there exist subsources $\tilde{X}$ of X and $\tilde{Y}$ of Y and a partition $\tilde{I}$ such that on these subsources $\tilde{I}$ is selected w.h.p. and $\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ is very close to uniform. As explained earlier this means that $\texttt{disp}$ is indeed a disperser.

8.3 The Proof

We start with proving Theorem 8.1. In Section 9 we give the additional details needed to prove Theorem 8.2. Throughout this section we assume that $\delta > 0$ and $m \geq 1$ are constants and that n is sufficiently large as a function of these two constants. In other words, all formal statements start with: "For every $\delta > 0$ and integer $m \geq 1$ there is an integer N_0 such that for every $n \geq N_0$ the following holds". We omit these clauses to make the reading easier. The proof closely follows the intuition explained earlier.

8.3.1 The main claim

We will prove that our disperser works by proving the following claim:

Claim 8.3. *For every X, Y independent random variables over $\{0,1\}^n$ with $H^\infty(X), H^\infty(Y) > \delta n$ there exist subsources $\tilde{X}$ of X and $\tilde{Y}$ of Y and a partition $\tilde{I}$ such that*

1. *$\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ is 2^{-10m}-close to uniform.*
2. *The partition $\tilde{I}$ is selected with probability $1 - 2^{-10m}$ on $x \leftarrow \tilde{X}$ and $y \leftarrow \tilde{Y}$.*

We now observe that Theorem 8.1 follows from Claim 8.3. The remainder of the section is devoted to proving Claim 8.3.

Proof of Theorem 8.1 from Claim 8.3 Let X, Y be independent δ-sources on n bits. Let $\tilde{X}, \tilde{Y}$ be the subsources guaranteed By Claim 8.3. Let $z \in \{0,1\}^m$ be some string. The first item of the claim gives that $\Pr[\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y}) = z] \geq 2^{-m} - 2^{-10m}$. The second item says that when $\texttt{disp}$ is applied on $\tilde{X}$ and $\tilde{Y}$ we have that $\tilde{I}$ is not selected with probability at most 2^{-10m}. In other words, $\Pr[\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y}) \neq \texttt{disp}(\tilde{X}, \tilde{Y})] \leq 2^{-10m}$. It follows that:

$$\Pr[\texttt{disp}(\tilde{X}, \tilde{Y}) = z] \geq \Pr[\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y}) = z] - 2^{-10m} \geq 2^{-m} - 2 \cdot 2^{-10m} > 0$$

We have that $\tilde{X} = (X|A)$ and $\tilde{Y} = (Y|B)$ for events $A, B \subseteq \{0,1\}^n$ and that $\Pr[X \in A]$ and $\Pr[Y \in B]$ are positive. Thus, when choosing $x \leftarrow X$ and $y \leftarrow Y$ we have positive probability of landing in $A \times B$ and in this case we have positive probability that $\texttt{disp}(x, y) = z$. Overall,

$$\Pr[\texttt{disp}(X, Y) = z] > 0$$

8.3.2 Nicely structured sources

When given arbitrary independent δ-sources X, Y our goal is to show the existence of subsources $\tilde{X}$ of X and $\tilde{Y}$ of Y and a partition $\tilde{I}$ with the two properties listed in Claim 8.3. The next Definition identifies properties of sources that will help us prove Claim 8.3.

Definition 8.4 (nicely structured sources)**.** Let $\tilde{X}, \tilde{Y}$ be random variables over $\{0,1\}^n$ and let $\tilde{I}$ be a partition. We say that $\tilde{X}, \tilde{Y}$ are *nicely structured sources* according to $\tilde{I}$ if $\tilde{X}, \tilde{Y}$ are independent and

- $H^\infty(\tilde{X}_1), H^\infty(\tilde{Y}_1) \geq \delta n/6$.
- $H^\infty(\tilde{X}_2), H^\infty(\tilde{Y}_2) \geq \delta^2 n/100$.
- $H^\infty(\tilde{X}_3), H^\infty(\tilde{Y}_3) = 0$ (or in other words that both $\tilde{X}_3, \tilde{Y}_3$ are fixed constants).

It is not important to remember the precise quantities in Definition 8.4. The important details are that in both sources the third segment is fixed while the two other segments have constant rate (where the constant is proportional to δ). Moreover, it is important to note that the entropy in the first segment ($\delta n/6$) is larger than the length of the second segment ($\delta n/10$). (Intuitively, this says that the first segment contains entropy even conditioned on the second segment).

The next Claim says that when $\texttt{disp}_{\tilde{I}}$ is applied on nicely structured sources it fulfils the first item of Claim 8.3 and produces a distribution that is close to uniform.

Claim 8.5. *Let $\tilde{X}$ and $\tilde{Y}$ be nicely structured sources according to a partition $\tilde{I}$. Then $\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ is 2^{-10m}-close to uniform.*

The proof of Claim 8.5 appears in Section 8.3.3. Loosely speaking, the claim follows as $\texttt{disp}_{\tilde{I}}(x, y)$ applies the extractor $\texttt{4ext}$ on the x_1, x_2, y_1, y_2 (arranged in the different order x_1, y_2, y_1, x_2). Although the four inputs are not independent, nicely structures sources are "sufficiently independent" for the argument to go through. (In fact, for Claim 8.5 we only need the first two requirements in Definition 8.4 and the third requirement will be used later on).

Another useful property of nicely structured sources is specified in the next Claim that says that any partition I that is smaller or incomparable to $\tilde{I}$ is likely to be rejected by the test.

Claim 8.6. *Let $\tilde{X}$ and $\tilde{Y}$ be nicely structured sources according to a partition $\tilde{I}$. Then:*

$$\Pr[\exists I \not\geq \tilde{I} : \ \texttt{test}_I(\tilde{X}, \tilde{Y}) = 1] \leq 2^{-10m}$$

The proof of Claim 8.6 appears in Section 8.3.4. Claim 8.6 means that in order to prove Claim 8.3 it is sufficient to show that for any independent δ-sources X, Y there exist subsources $\tilde{X}$ of X and $\tilde{Y}$ of Y which are nicely structured and in addition $\Pr[\texttt{test}_{\tilde{I}}(\tilde{X}, \tilde{Y}) = 1] = 1$. In such subsources the "correct partition" $\tilde{I}$ is selected with high probability. This is because $\tilde{I}$ is a unique minimal partition that passes the test.

The plan is to first show that for every two independent δ-sources X, Y there exist subsources X' of X and Y' of Y which are nicely structured (according to some partition $\tilde{I}$) and then we will

show the existence of subsources $\tilde{X}$ of X' and $\tilde{Y}$ of Y' which in addition to being nicely structured have the additional property above. As the second step of restricting X' and Y' to subsources $\tilde{X}$ and $\tilde{Y}$ "loses entropy" we phrase the next Claim using entropy requirements that are slightly larger than those made in Definition 8.4. This will allow us to lose entropy in the second step and still obtain sources that are nicely structured. The following Claim implements the first step above.

Claim 8.7. *For every X, Y independent random variables over $\{0,1\}^n$ with $H^\infty(X), H^\infty(Y) > \delta n$ there exist subsources X' of X and Y' of Y and a partition $\tilde{I}$ such that when partitioning according to $\tilde{I}$.*

- $H^\infty(X'_1), H^\infty(Y'_1) \geq \delta n/2$.
- $H^\infty(X'_2), H^\infty(Y'_2) \geq \delta^2 n/50$.
- $H^\infty(X'_3), H^\infty(Y'_3) = 0$ *(or in other words that both X'_3, Y'_3 are fixed constants).*

The Proof of Claim 8.7 appears in Section 8.3.5. The second step in the plan above is implemented in the following Claim.

Claim 8.8. *Let $\tilde{I}$ be a partition and let X', Y' be independent distributions which satisfy the guarantee of Claim 8.7. Namely, when partitioning according to $\tilde{I}$.*

- $H^\infty(X'_1), H^\infty(Y'_1) \geq \delta n/2$.
- $H^\infty(X'_2), H^\infty(Y'_2) \geq \delta^2 n/50$.
- $H^\infty(X'_3), H^\infty(Y'_3) = 0$

Then, there exist subsources $\tilde{X}$ of X' and $\tilde{Y}$ of Y' such that:

- *$\tilde{X}, \tilde{Y}$ are nicely structured according to $\tilde{I}$.*
- $\Pr[\texttt{test}_{\tilde{I}}(\tilde{X}, \tilde{Y}) = 1] = 1$.

The proof of Claim 8.8 appears in Section 8.3.6. Putting everything together allows us to prove Claim 8.3 (which in turn implies Theorem 8.1):

Proof of Claim 8.3 Let X, Y be independent δ-sources. By the combination of Claim 8.7 and Claim 8.8 we get that there exists a partition $\tilde{I}$ and subsources $\tilde{X}$ of X and $\tilde{Y}$ of Y which are nicely structured according to $\tilde{I}$ and furthermore, $\Pr[\texttt{test}_{\tilde{I}}(\tilde{X}, \tilde{Y}) = 1] = 1$. The first item of Claim 8.3 follows directly from Claim 8.5. For the second item we note that by Claim 8.6:

$$\Pr[\exists I \not\succeq \tilde{I} : \ \texttt{test}_I(\tilde{X}, \tilde{Y}) = 1] \leq 2^{-10m}$$

Whenever the event above does not hold we have that $\tilde{I}$ is the partition that is selected and thus, $\tilde{I}$ is selected with probability at least $1 - 2^{-10m}$ as required.

8.3.3 Extractor for nicely structured sources: proof of Claim 8.5

We now prove Claim 8.5. Recall that $\mathtt{disp}_I(x,y) = \mathtt{opt}(\mathtt{s_ext}_{d/\ell}(x_1,y_2), \mathtt{s_ext}_{d/\ell}(y_1,x_2))$. We will show that for most fixings x_2 of $\tilde{X}_2$ and y_2 of $\tilde{Y}_2$ the 2-source extractor $\mathtt{opt}$ is applied on independent distributions with sufficient min-entropy. It follows that for most fixings the output of $\mathtt{opt}$ is close to uniform.

We first apply Lemma 3.3 on the sources $\tilde{X}_1, \tilde{X}_2$ setting $\rho = 2^{-\delta n/100}$ we conclude that with probability $1-\rho$ over choosing $x_2 \leftarrow \tilde{X}_2$,

$$H^\infty(\tilde{X}_1|\tilde{X}_2 = x_2) \geq \delta n/6 - \delta n/10 - \log(1/\rho) > \delta n/200$$

We say that x_2 is "useful" if the inequality above holds for x_2.

The function $\mathtt{s_ext}_{d/\ell}$ is a strong somewhere random extractor with error $2^{-\eta n}$. Recall that a string w is good for a distribution R if $\mathtt{s_ext}_{d/\ell}(R,w)$ is $2^{-\eta n}$-close to a somewhere random distribution. As $\mathtt{s_ext}$ can handle sources with min-entropy $\delta n/200$ we have that for any two independent distributions R, W both with min-entropy at least $\delta n/200$,

$$\Pr_{w \leftarrow W}[w \text{ is good for } R] \geq 1 - 2^{-\eta n}$$

We say that a string y_2 is "extracting" with respect to x_2 if y_2 is good for the distribution $(\tilde{X}_1|\tilde{X}_2 = x_2)$. It follows that for every useful x_2,

$$\Pr[\tilde{Y}_2 \text{ is extracting with respect to } x_2] \geq 1 - 2^{-\eta n}$$

As this holds for every useful x_2 and as we know that the weight of useful strings x_2 in the distribution $\tilde{X}_2$ is $1-\rho$ we conclude that:

$$\Pr[\tilde{Y}_2 \text{ is extracting with respect to } \tilde{X}_2] \geq 1 - 2^{-\eta n} - \rho$$

We repeat the argument replacing the roles of x and y and define the notions of "useful" and "extracting" strings for this setup. (To distinguish these new notions from the previously defined notions we denote the notions "useful*" and "extracting*". More precisely, we say that a string y_2 is useful* if $H^\infty(\tilde{Y}_1|\tilde{Y}_2 = y_2) > \delta n/200$ and we say that a string x_2 is extracting* for y_2 if x_2 is good for the distribution $(\tilde{Y}_1|\tilde{Y}_2 = y_2)$. The same argument as above gives that:

$$\Pr[\tilde{X}_2 \text{ is extracting* with respect to } \tilde{Y}_2] \geq 1 - 2^{-\eta n} - \rho$$

Thus, by a union bound with probability $1 - 2 \cdot (2^{-\eta n} + 2\rho)$ over choosing $(x_2, y_2) \leftarrow (\tilde{X}_2, \tilde{Y}_2)$ the pair that is chosen is a pair of elements that are a "good match" that is y_2 is extracting with respect to x_2 and x_2 is extracting* with respect to y_2.

Plan for the rest of the proof Next we show that for any pair (x_2, y_2) that is a good match $\mathtt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ is close to uniform when conditioning on the event $E = \{\tilde{X}_2 = x_2 \text{ and } \tilde{Y}_2 = y_2\}$. It will follow that $\mathtt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ is close to uniform.

We now implement the outline above. For a fixed pair (x_2, y_2) that is a good match we consider the event $E = \{\tilde{X}_2 = x_2 \text{ and } \tilde{Y}_2 = y_2\}$ and the conditional distribution: $(X', Y') = ((\tilde{X}, \tilde{Y})|E)$. We define:

- $Z_1 \stackrel{def}{=} \mathtt{s_ext}_{d/\ell}(X'_1, Y'_2) = \mathtt{s_ext}_{d/\ell}((\tilde{X}_1|\tilde{X}_2 = x_2), y_2)$.
- $Z_2 \stackrel{def}{=} \mathtt{s_ext}_{d/\ell}(Y'_1, X'_2) = \mathtt{s_ext}_{d/\ell}((\tilde{Y}_1|\tilde{Y}_2 = y_2), x_2)$.

Note that Z_1 and Z_2 are independent (as each of them depends only on one of the sources). Furthermore, by the fact that (x_2, y_2) is a good match we have that both distributions are $2^{-\eta n}$-close to somewhere random distributions. A somewhere random distribution with ℓ blocks of length d/ℓ is of length d and by Lemma 4.8 such a distribution has min-entropy at least $d/\ell - \log \ell$. We have required that $d/\ell - \log \ell \geq 10 \log d$ and therefore such distributions meet the entropy requirements of $\texttt{opt}$. As the error of $\texttt{opt}$ is 2^{-100m}, it follows that $\texttt{opt}(Z_1, Z_2)$ is $(2 \cdot 2^{-\eta n} + 2^{-100m})$-close to uniform.

Summing up, when applying $\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ with probability $1 - 2 \cdot (2^{-\eta n} + \rho)$ we obtains strings x_2, y_2 which are a good match and in this case the output of $\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ is $(2 \cdot 2^{-\eta n} + 2^{-100m})$-close to uniform. Thus, for large enough n so that $2^{-\eta n} + \rho \ll 2^{-100m}$ we have that $\texttt{disp}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ is 2^{-10m}-close to uniform as required.

8.3.4 Rejecting incorrect partitions: proof of Claim 8.6

We now prove Claim 8.6. Let I be a partition such that $I' \not\succeq \tilde{I}$. We want to show that $\texttt{test}_I(\tilde{X}, \tilde{Y})$ accepts with very small probability so that we can do a union bound over all such partitions I and get that all of them are rejected simultaneously with high probability.

Let $\tilde{I} = (\tilde{i}_1, \tilde{i}_2)$ and let $I = (i_1, i_2)$. As $I' \not\succeq \tilde{I}$ we assume without loss of generality that $i_1 < \tilde{i}_1$ (the proof is similar in the case $i_2 < \tilde{i}_2$). We know that $H^\infty(\tilde{X}_2^{\tilde{I}}) \geq \delta^2 n/100$ (or in words that the second segment according to $\tilde{I}$ contains randomness). Note that as $i_1 \leq \tilde{i}_1$ the partition according to I puts the second segment of $\tilde{I}$ in the third segment of I, and therefore

$$H^\infty(\tilde{X}_3^I) \geq \delta^2 n/100$$

From now on we use the partition I to divide the input strings in all expressions below. We need to show that w.h.p. the challenge string $c = c(\tilde{X}, \tilde{Y})$ is not a sub-block of any response string $c_{v_1,v_2}(\tilde{X}, \tilde{Y})$. Fix some $(v_1, v_2) \in [p]^2$. We apply Lemma 3.3 on the variables $\tilde{X}_3$ and $\tilde{X}[v_1]$ using $\rho = 2^{-\delta^3 n}$ and conclude that with probability $1 - 2^{-\delta^3 n}$ over choosing $\hat{x} \leftarrow \tilde{X}[v_1]$

$$H^\infty(\tilde{X}_3 | \tilde{X}[v_1] = \hat{x}) \geq \delta^2 n/100 - 2\delta^3 n \geq \delta^2 n/200$$

We say that a string $\hat{x}$ of length $\delta^3 n$ is "nice" if it satisfies the inequality above.

We can apply the same argument on $\tilde{Y}$ and $\tilde{Y}[v_2]$ and obtain that with probability $1 - 2^{-\delta^3 n}$ over choosing $\hat{y} \leftarrow \tilde{Y}[v_2]$

$$H^\infty(\tilde{Y} | \tilde{Y}[v_2] = \hat{y}) \geq \delta^2 n/200$$

(Note that here we use all of $\tilde{Y}$ and not just $\tilde{Y}_3$. This is because c_1 is a function of x_3 and y). We say that a string $\hat{y}$ of length $\delta^3 n$ is "nice*" if it satisfies the inequality above.

Plan for the rest of the proof We will show that for any nice $\hat{x}$ and nice* $\hat{y}$ the probability that the challenge string $c(\tilde{X}, \tilde{Y})$ appears as a sub-block in the response string $c_{v_1,v_2}(\tilde{X}, \tilde{Y})$ is small when conditioned on the event $\{\tilde{X}[v_1] = \hat{x} \text{ and } \tilde{Y}[v_2] = \hat{y}\}$. It follows that the probability that $c(\tilde{X}, \tilde{Y})$ appears as a sub-block in the response string $c_{v_1,v_2}(\tilde{X}, \tilde{Y})$ is small overall. By a union bound over all pairs (v_1, v_2) it will follow that $\texttt{test}_I(\tilde{X}, \tilde{Y})$ rejects with high probability. This is because w.h.p. the challenge string c is not responded by any response string c_{v_1,v_2}.

We now implement the plan above. Fix some nice $\hat{x}$ and nice* $\hat{y}$ and let E be the event $\{\tilde{X}[v_1] = \hat{x} \text{ and } \tilde{Y}[v_2] = \hat{y}\}$. We consider the distribution $(X', Y') = ((\tilde{X}, \tilde{Y})|E)$. Recall that $c_{v_1,v_2}(x, y) = \texttt{s_ext}_{2\ell k}(x[v_1], y[v_2])$. Thus, $c_{v_1,v_2}(X', Y') = \texttt{s_ext}_{2\ell k}(\hat{x}, \hat{y})$ is a fixed string. On the other hand, recall that $c_1(x, y) = \texttt{s_ext}_k(x_3, y)$ and therefore $c_1(X', Y') = \texttt{s_ext}_k(X'_3, Y')$. Note

that X'_3, Y' are independent distributions where each has min-entropy at least $\delta^2 n/200$. Thus, they meet the entropy requirement of $\mathtt{s_ext}_k$ and $c_1(X', Y')$ is $2^{-\eta n}$-close to a somewhere random distribution. By Lemma 4.8 a somewhere random distribution on ℓ blocks of length k has min-entropy at least $k - \log \ell$. Therefore, the challenge string $c(X', Y')$ (which contains $c_1(X', Y')$ as a substring) is $2^{-\eta n}$-close to having min-entropy $k - \log \ell$. It indeed follows that it is very unlikely that the challenge string $c(X', Y')$ is a sub-block of the fixed response string $c_{v_1,v_2}(X', Y')$. More precisely, let a be any fixed string that is composed of ℓ blocks of length $2k$. By a union bound over the ℓ blocks the probability that a random variable B with min-entropy v appears as a sub-block of a is bounded by $\ell 2^{-v}$. Thus, it follows that the probability that $c(X', Y')$ appears as a sub-block of $c_{v_1,v_2}(X', Y')$ is at most $\ell \cdot (2^{-(k-\log \ell)} + 2^{-\eta n}) \leq \ell^3 2^{-k}$ for large enough n. In words, the probability that the challenge c is responded by c_{v_1,v_2} is small.

Summing up, for any $(v_1, v_2) \in [p]^2$ we have that with probability $1 - 2 \cdot 2^{-\delta^3 n}$ over the choice of $\tilde{X}[v_1], \tilde{Y}[v_2]$ we obtain strings $\hat{x}, \hat{y}$ which are nice and then the probability that the challenge string c appears as a sub-block in the response string c_{v_1,v_2} is at most $\ell^2 2^{-k}$. Thus, overall the probability (over $\tilde{X}, \tilde{Y}$) that this event happens is at most $2 \cdot 2^{-\delta^3 n} + \ell^3 2^{-k} \leq \ell^4 2^{-k}$ for large enough n.

This holds for one fixed partition and fixed pair $(v_1, v_2) \in [p]^2$. However, by a union bound over all partitions $I \not\succeq \tilde{I}$ and all pairs (v_1, v_2) we have that the challenge string of the partition I is not responded by any of the response strings with probability at least $1 - p^2 t^2 \ell^4 2^{-k}$. We have chosen the constant k so that this quantity is at least $1 - 2^{-10m}$ as required.

8.3.5 Existence of nicely structured sources: proof of Claim 8.7

We now prove Claim 8.7. It is sufficient to consider the sources X, Y one at a time. We will show that there exists a subsource X' of X and an index $i_1 \in [t]$ such that when partitioning according to i_1 we have that:

- $H^\infty(X'_1) \geq \delta n/2$.
- $H^\infty(X'_2) \geq \delta^2 n/50$.
- $H^\infty(X'_3) = 0$.

We can then complete the proof of the Claim by doing the same argument on Y to obtain a subsource Y' and an index i_2 with the same properties and then setting $\tilde{I} = (i_1, i_2)$

We now turn our attention to finding a subsource X' with the aforementioned properties. We show the existence of this subsource using an iterative process: We set $i = t$ and $X' = X$. At each step of the process we decrease i by one and modify X'. We will show that at some point we obtain an index i and a subsource X' such that partitioning X' according to $i_1 = i$ gives the desired properties. During the process we maintain the invariant that X' is a subsource of X, and when partitioning according to i, X'_3 is fixed and $H^\infty(X'_1, X'_2) \geq 3\delta n/4 + i\delta^2 n/50$. Note that this indeed holds at the beginning of the process as X'_3 is of length zero and $3\delta n/4 + t\delta^2 n/50 \leq \delta n$. In each iteration there are two possibilities:

- If $H^\infty(X'_2) \geq \delta^2 n/50$ then we stop the process and set $i_1 = i$. Note that in this case X' is a subsource of X with the required properties. That is $H^\infty(X'_3) = 0$, $H^\infty(X'_2) \geq \delta^2 n/50$ and by Lemma 3.2 $H^\infty(X'_1) \geq H^\infty(X'_1, X'_2) - |X'_2| \geq 3\delta n/4 - \delta n/10 \geq \delta n/2$.
- If $H^\infty(X'_2) < \delta^2 n/50$ then there exists a string x_2 of length $\delta n/10$ so that $\Pr[X'_2 = x_2] > 2^{-\delta^2 n/50}$. We consider the subsource $X'' = (X'|X'_2 = x_2)$. By Lemma 3.1 we have that $H^\infty(X'') \geq H^\infty(X') - \delta^2 n/50 \geq H^\infty(X'_1, X'_2) - \delta^2 n/50 \geq 3\delta n/4 + (i-1)\delta^2 n/50$. As X''_2, X''_3

are fixed all this entropy lies in X_1''. Note that if we now partition X'' according to $i-1$ we have that X_3'' is fixed and $H^\infty(X_1'', X_2'') \geq 3\delta n/4 + (i-1)\delta^2 n/50$. Thus, we can set X' to X'' and i to $i-1$ while maintaining the invariant and continue the process.

We note that the process above must stop before we reach $i = 1$. This is because for $i = 1$ we have that $i\delta n/10 < 3\delta n/4$ and it is impossible for the invariant to hold as when partitioning according to i the length of the first and second segment is $\delta n/10$ and it is impossible that these segments have min-entropy $3\delta n/4$. It follows that we indeed find a subsource with the required properties.

8.3.6 Selecting the correct partition: proof of Claim 8.8

We now prove proof of Claim 8.8. We have independent distributions X', Y' which are already nicely structured according to $\tilde{I}$ and our goal is to restrict them to subsources so that the additional requirement that $\texttt{test}_{\tilde{I}}$ passes holds without reducing the entropy in any of the segments by much so that the subsources we obtain are still nicely structured.

Fixing the challenge. Recall that $c_1(x, y) = \texttt{s_ext}_k(x_3, y)$ and $c_2(x, y) = \texttt{s_ext}_k(y_3, x)$. Our next goal is to restrict our attention to subsources in which c_1 and c_2 are fixed. (Recall that c_1 is already fixed if x_3 is of length zero and c_2 is fixed if y_3 is of length zero). We start with fixing c_2 in the case that y_3 has length greater than zero. We have that Y_3' is fixed to some value y_3' and therefore the value of $c_2(X', Y')$ given by $\texttt{s_ext}(y_3', X')$ depends only on X'. Let c_2' be the most likely value of $\texttt{s_ext}(y_3', X')$ and let E be the event $\{\texttt{s_ext}(Y_3', X') = c_2'\}$ as the string c_2 is k bit long we have that $\Pr[E] \geq 2^{-k}$. Recall that k is a constant that depends only on δ and m. We define $X'' = (X'|E)$ and note that X'' is a subsource of measure 2^{-k} of X. Note that the distributions X_1'' and X_2'' can be expressed as $(X_2'|E)$ and $(X_1'|E)$. (Note that we are not claiming that these are subsources of X_1' and X_2'). By Lemma 3.1 it follows that the conditioning loses at most k bits of min-entropy. We conclude that $H^\infty(X_1'') \geq H^\infty(X_1') - k$ and $H^\infty(X_2'') \geq H^\infty(X_2') - k$. As k is a constant this means that the amount of min-entropy in each of the two segments (which was initially large) did not drop significantly.

We repeat the same process and fix $c_1(X', Y')$ to some value c_1' by restricting our attention to a subsource Y'' of Y'. Thus, in X'', Y'' the value of $c = c_1 \circ c_2$ is fixed to some value c'. As k is a constant we have that for large enough n the independent sources X'', Y'' have the following properties:

- $H^\infty(X_1''), H^\infty(Y_1'') \geq \frac{\delta n}{4}$.
- $H^\infty(X_2''), H^\infty(Y_2'') \geq \frac{\delta^2 n}{60}$.
- $H^\infty(X_3''), H^\infty(Y_3'') = 0$ (as these segments were already fixed in X', Y').
- On the subsources X'', Y'' the value of the "challenge string" c is fixed. More formally, $\Pr[c(X'', Y'') = c'] = 1$.

Finding a good pair We have that $H^\infty(X'') \geq H^\infty(X_1'') \geq \delta n/4$. We consider the partition of X'' into $X''[1], \ldots, X''[p]$. Intuitively, we can expect that at least one of these segments has min-entropy at least $H^\infty(X'')/p$. While this is not necessarily true, the next Lemma asserts that there exists a large subsource of X'' which has a high min-entropy block.

Claim 8.9. *There exist a subsource X''' of X'' of measure $1/p$ and a $v_1 \in [p]$ such that $H^\infty(X'''[v_1]) \geq \delta^4 n/20$.*

Proof. (of Claim 8.9) To simplify the notation in this proof let $W = X''$ and we denote $W[v]$ by W_v and let $W_{>v}$ denote the concatenation of $W_{v+1}, \ldots, W_p$. We have that $H^\infty(W) \geq \delta n/4$. For every $w \in \{0,1\}^n$ that is in the support of W and $v \in [p]$ we define:

$$r_v(w) = \Pr[W_v = w_v | W_{>v} = w_{>v}]$$

Note that for any such w,

$$2^{-\delta n/4} \geq \Pr[W = w] = \prod_{v \in [p]} r_v(w)$$

It follows that for every such w there exists $v \in [p]$ such that $r_v(w) \leq 2^{-\delta n/4p} = 2^{-\delta^4 n/4}$. For every w we denote the largest such v by $v(w)$. (It is not important to choose the largest v and we only need to associate one v with any string w). Let $A_v = \{w : v(w) = v\}$. Note that the sets A_v are a partition of the support of W and thus there exists $v \in [p]$ such that $\Pr[W \in A_v] \geq 1/p \geq \delta^3$. To simplify the notation we denote this set by A.

Let $W' = (W|W \in A)$ be a subsource of W. We now show that $H^\infty(W_v) \geq \delta^4 n/20$. This indeed suffices to prove the Claim.

Let $u = \delta^3 n(p-v)$ denote the total length of the blocks $v+1, \ldots p$. Fix some string $\hat{w}$ of length $\delta^3 n$ we have that:

$$\begin{aligned}
\Pr[W'_v = \hat{w}] &= \Pr[W_v = \hat{w} | W \in A] \\
&= \frac{\Pr[W_v = \hat{w} \wedge W \in A]}{\Pr[W \in A]} \\
&\leq \frac{\Pr[W_v = \hat{w} \wedge W \in A]}{\delta^3} \\
&= \frac{\sum_{y \in \{0,1\}^u} \Pr[W_v = \hat{w} \wedge W \in A | W_{>v} = y] \cdot \Pr[W_{>v} = y]}{\delta^3}
\end{aligned}$$

For every $\hat{w}$ let $A_{\hat{w}} = \{y : \Pr[W_v = \hat{w} | W_{>v} = y] \leq 2^{-\delta^4 n/4}\}$. For $y \notin A_{\hat{w}}$ we have that

$$\Pr[W_v = \hat{w} \wedge W \in A | W_{>v} = y] = 0$$

This is because for any string $w \in A$, $w_{>v} \in A_{w_v}$. Thus, the only positive entries in the sum above correspond to $y \in A_{\hat{w}}$.

$$\begin{aligned}
&= \frac{\sum_{y \in A_{\hat{w}}} \Pr[W_v = \hat{w} \wedge W \in A | W_{>v} = y] \cdot \Pr[W_{>v} = y]}{\delta^3} \\
&\leq \frac{\sum_{y \in A_{\hat{w}}} \Pr[W_v = \hat{w} | W_{>v} = y] \cdot \Pr[W_{>v} = y]}{\delta^3} \\
&\leq \frac{\sum_{y \in A_{\hat{w}}} 2^{-\delta^4 n/4} \cdot \Pr[W_{>v} = y]}{\delta^3} \qquad \text{(By the definition of } A_{\hat{w}}\text{)} \\
&\leq \frac{2^{-\delta^4 n/4}}{\delta^3} \\
&\leq 2^{-\delta^4 n/20}
\end{aligned}$$

□

Note that $1/p$ is a constant that depends only on δ and therefore when restricting X'' to X''' we only lose a constant amount of min-entropy. As we did previously, we can apply this argument

separately for X_1''' and X_2''' and get that each of them suffers a loss of at most a constant amount of min-entropy compared to X_1'', X_2''.

We repeat the same process for Y'' to show the existence of a subsource Y''' and $v_2 \in [p]$ with analogous properties. Thus, the subsources X''', Y''' have the following properties:

- $H^\infty(X_1'''), H^\infty(Y_1''') \geq \frac{\delta n}{5}$.
- $H^\infty(X_2'''), H^\infty(Y_2''') \geq \frac{\delta^2 n}{70}$.
- $H^\infty(X_3'''), H^\infty(Y_3''') = 0$ (as these segments were already fixed in X', Y').
- On the subsources X''', Y''' the value of the "challenge string" c is fixed. More formally, $\Pr[c(X'', Y'') = c'] = 1$.
- There exist $(v_1, v_2) \in [p]^2$ such that $H^\infty(X'''[v_1]), H^\infty(Y'''[v_2]) \geq \frac{\delta^4 n}{20}$.

Responding to the challenge. Recall that for every pair $(v_1, v_2) \in [p]^2$ the construction computes a "response string" $c_{v_1,v_2}(x, y) = \texttt{s_ext}_{2\ell k}(x[v_1], x[v_2])$ and checks whether it "meets the challenge" in the sense that it has the "challenge string" $c(x, y)$ as a sub-block.

We now further restrict the subsources so that the response string meets the challenge. We have that for the pair (v_1, v_2) found earlier both the substrings $X'''[v_1]$ and $Y'''[v_2]$ have constant min-entropy rate and are independent. It follows, that when applying $\texttt{s_ext}_{2\ell k}(X'''[v_1], Y'''[v_2])$ we obtain a distribution $\mathcal{R}$ that is $2^{-\eta n}$-close to a somewhere random distribution. This distribution has blocks of length $2\ell k$ and note that this is a constant that depends only on δ and m. We have that there exists a random variable J over $[\ell]$ (that may depend on $\mathcal{R}$) such that $\mathcal{R}_J$ is $2^{-\eta n}$ close to uniform. Thus, $\mathcal{R}_J$ equals the fixed challenge string c' with probability at least $2^{-|c'|} - 2^{-\eta n}$ which is larger than some positive constant. Thus, with constant positive probability the challenge c' is responded.

We say that a pair of strings $\hat{x}, \hat{y}$ each of length $\delta^3 n$ is "useful" if

- $\texttt{s_ext}_{2\ell k}(\hat{x}, \hat{y})$ contains the challenge c' as a sub-block.
- $\Pr[X'''[v_1] = \hat{x}] > 2^{-2\delta^3 n}$ and $\Pr[Y'''[v_2] = \hat{y}] > 2^{-2\delta^3 n}$.

We have already seen that the probability that $X'''[v_1], X'''[v_2]$ fulfill the first item is larger than some positive constant. Let $B_1 = \{\hat{x} : \Pr[X'''[v_1] = \hat{x}] \leq 2^{-2\delta^3 n}\}$. Note that

$$\Pr[X'''[v_1] \in B_1] \leq 2^{\delta^3 n} \cdot 2^{-2\delta^3 n} = 2^{-\delta^3 n}$$

The same argument applies to the second source and we conclude that for large enough n the probability that $X'''[v_1], Y'''[v_2]$ are useful is larger than some positive constant.

Fix some useful strings $\hat{x}, \hat{y}$. We define subsources $\tilde{X}, \tilde{Y}$ of X''', Y''' as follows: $\tilde{X} = (X'''|X'''[v_1] = \hat{x})$ and $\tilde{Y} = (Y'''|Y'''[v_2] = \hat{y})$. Note that $\tilde{X}, \tilde{Y}$ are indeed subsources. (These are going to be the final subsources). Note that $c_{v_1,v_2}(\tilde{X}, \tilde{Y})$ is now fixed to the constant $\texttt{s_ext}_{2\ell k}(\hat{x}, \hat{y})$ which in turn contains the fixed challenge string c' as a sub-block (as $\hat{x}, \hat{y}$ are useful). Thus, in $\tilde{X}, \tilde{Y}$ the challenge is responded with probability one and $\Pr[\texttt{test}_{\tilde{I}}(\tilde{X}, \tilde{Y}) = 1] = 1$. Furthermore, by Lemma 3.1, when moving from X''' to $\tilde{X}$, the min-entropy of the first and second segments decreases by at most $2\delta^3 n$ bits. The same argument applies to the second source. We can assume w.l.o.g. that δ is sufficiently small so that δ^3 is much smaller than δ^2 and so the losses in min-entropy in each of the segments are insignificant. It follows that the two sources $\tilde{X}, \tilde{Y}$ are nicely structured. More precisely, we have that:

- $H^\infty(\tilde{X}_1), H^\infty(\tilde{Y}_1) \geq \frac{\delta n}{6}$.
- $H^\infty(\tilde{X}_2), H^\infty(\tilde{Y}_2) \geq \frac{\delta^2 n}{100}$.
- $H^\infty(\tilde{X}_3), H^\infty(\tilde{Y}_3) = 0$.
- $\texttt{test}_{\tilde{I}}(\tilde{X}, \tilde{Y})$ accepts with probability one.

The Claim follows.

Extending the proof of Claim 8.8 to obtain "large" subsources So far we did not pay attention to the measure of the subsources $\tilde{X}, \tilde{Y}$ and now we would like to say that they have large measure (say some positive constant). Following the argument above more carefully we notice that X'' has measure 2^{-k} as a subsource of X' and X''' has measure $1/p$ as a subsource of X''. Thus, overall the measure of X''' as a subsource of X' is a positive constant. Unfortunately, the measure of $\tilde{X}$ as a subsource of X''' is $2^{-2\delta^3 n}$ which is smaller than a constant. While the argument above does not give that $\tilde{X}$ has constant measure as a subsource of X' it gives a weaker statement with the same flavor. Namely, under the assumptions of Claim 8.8 there is a subsource collection $\tilde{X}_1, \ldots, \tilde{X}_w$ and $\tilde{Y}_1, \ldots, \tilde{Y}_w$ of (X', Y') (for some integer w) such that the subsource collection has a measure that is a positive constant and for every $j \in [w]$, $\tilde{X}_j, \tilde{Y}_j$ have the properties guaranteed by Claim 8.8. The precise statement appears below:

Claim 8.10. *There exists a constant $\mu > 0$ (that depends only on δ and m) such that the following holds: Let $\tilde{I}$ be a partition and let X', Y' be independent distributions which satisfy the guarantee of Claim 8.7. Namely, when partitioning according to $\tilde{I}$.*

- $H^\infty(X'_1), H^\infty(Y'_1) \geq \delta n/2$.
- $H^\infty(X'_2), H^\infty(Y'_2) \geq \delta^2 n/50$.
- $H^\infty(X'_3), H^\infty(Y'_3) = 0$

Then there is an integer w and a subsource collection $\tilde{X}_1, \ldots, \tilde{X}_w$ and $\tilde{Y}_1, \ldots, \tilde{Y}_w$ of the distribution (X', Y') that has measure μ and for every $j \in [w]$, $\tilde{X}_w, \tilde{Y}_w$ has the properties guaranteed by Claim 8.8, namely:

- *$\tilde{X}_w, \tilde{Y}_w$ are nicely structured according to $\tilde{I}$.*
- $\Pr[\texttt{test}_{\tilde{I}}(\tilde{X}_w, \tilde{Y}_w) = 1] = 1$.

Proof of Claim 8.10 We explain how this follows from the proof of Claim 8.8. We have that X''' is a subsource of X' of measure $2^{-k}/p$ and Y''' is a subsource of Y' of measure $2^{-k}/p$. Recall that $2^{-k}/p$ is a positive constant that depends only on δ and m. Recall that to construct $\tilde{X}$ and $\tilde{Y}$ as subsources of the sources X''' and Y''' we defined $\tilde{X} = (X'''|X'''[v_1] = \hat{x})$ and $\tilde{Y} = (Y'''|Y'''[v_2] = \hat{y})$ where $\hat{x}, \hat{y}$ are *any* useful strings. We have shown that the probability that $X'''[v_1]$ and $Y'''[v_2]$ are useful is larger than a positive constant. Let w be the number of pairs $(\hat{x}, \hat{y})$ of useful strings and enumerate all these pairs in some order. That is, for $j \in [w]$ we use $(\hat{x}_j, \hat{y}_j)$ to denote the j'th pair of useful strings. For every $j \in [w]$ let $\tilde{X}_j = (X'''|X'''[v_1] = \hat{x}_j)$ and $\tilde{Y}_j = (Y'''|Y'''[v_1] = \hat{y}_j)$. Note that the collection $\tilde{X}_1, \ldots, \tilde{X}_w$ and $\tilde{Y}_1, \ldots, \tilde{Y}_w$ are a subsource collection of (X''', Y''') of measure which is some positive constant. As X''' is a subsource of X' of constant measure we have by Fact 4.5 that the collection is also a subsource collection of (X', Y') of positive constant measure. For any $j \in [w]$ the distributions $\tilde{X}_j, \tilde{Y}_j$ were defined using useful strings and we have already seen that they satisfy the guarantee of Claim 8.8.

9 Proof for Stronger Notion of Dispersers

We now explain how to use the machinery we developed in order to prove Theorem 8.2. We imitate the argument used in the proof of Theorem 8.1 making some changes. We start by stating a stronger notion of Claim 8.3. Intuitively, we want that for every independent δ-sources X, Y there exist *large* subsources $\tilde{X}, \tilde{Y}$ with the properties guaranteed in Claim 8.3. The stronger theorem would have followed had we been able to prove that there exist subsources with measure that is some positive constant. While we do not know how to prove this it will suffice to show that there exists a subsource collection $\tilde{X}_1, \ldots, \tilde{X}_w$ and $\tilde{Y}_1, \ldots, \tilde{Y}_w$ (for some integer w) that is of constant measure and for each $j \in [w]$, $\tilde{X}_j, \tilde{Y}_j$ have the properties guaranteed by Claim 8.3. The following Claim is a stronger version of Claim 8.3.

Claim 9.1. *There exists a constant $\mu > 0$ (that depends only on δ and m) such that the following holds: For every X, Y independent random variables over $\{0,1\}^n$ with $H^\infty(X), H^\infty(Y) > \delta n$ there exists a partition $\tilde{I}$, an integer w and a subsource collection $\tilde{X}_1, \ldots, \tilde{X}_w$ and $\tilde{Y}_1, \ldots, \tilde{Y}_w$ of (X, Y) that has measure μ such that for every $j \in [w]$*

1. $\texttt{disp}_{\tilde{I}}(\tilde{X}_j, \tilde{Y}_j)$ *is 2^{-10m}-close to uniform.*
2. *The partition $\tilde{I}$ is selected with probability $1 - 2^{-10m}$ on $x \leftarrow \tilde{X}_j$ and $y \leftarrow \tilde{Y}_j$.*

Proof of Theorem 8.2 from Claim 9.1 The proof is identical to that of Theorem 8.1 when using Claim 8.3 (except that we now have a subsource collection and not a subsource). The argument follows: Let X, Y be independent δ-sources on n bits. Let $\tilde{I}$ be a partition and $\tilde{X}_1, \ldots, \tilde{X}_w$ and $\tilde{Y}_1, \ldots, \tilde{Y}_w$ be a subsource collection guaranteed By Claim 9.1. Let $z \in \{0,1\}^m$ be some string. The first item of the claim gives that for any $j \in [w]$, $\Pr[\texttt{disp}_{\tilde{I}}(\tilde{X}_j, \tilde{Y}_j) = z] \geq 2^{-m} - 2^{-10m}$. The second item says that when $\texttt{disp}$ is applied on $\tilde{X}_j$ and $\tilde{Y}_j$ we have that $\tilde{I}$ is not selected with probability at most 2^{-10m}. In other words, $\Pr[\texttt{disp}_{\tilde{I}}(\tilde{X}_j, \tilde{Y}_j) \neq \texttt{disp}(\tilde{X}_j, \tilde{Y}_j)] \leq 2^{-10m}$. It follows that:

$$\Pr[\texttt{disp}(\tilde{X}_j, \tilde{Y}_j) = z] \geq \Pr[\texttt{disp}_{\tilde{I}}(\tilde{X}_j, \tilde{Y}_j) = z] - 2^{-10m} \geq 2^{-m} - 2 \cdot 2^{-10m} > 2^{-(m+1)}$$

Let $A_1, \ldots, A_w$ and $B_1, \ldots, B_w$ be defining events for the subsource collection. Let $E_j = A_j \times B_j$ and let $E = \cup_j E_j$. We have that the events $E_1, \ldots, E_w$ are a partition of E and that $\Pr[(X, Y) \in E] \geq \mu$. Once we land in E we have already shown that we hit z with probability $2^{-(m+1)}$ and so overall, $\Pr[\texttt{disp}(X, Y) = z] \geq \mu 2^{-(m+1)}$ and note that this quantity is a constant that depends only on δ and m as required.

9.1 Proof of Claim 9.1

We follow the proof of Claim 8.3. That proof worked in two steps. We first used Claims 8.7 and Claim 8.8 to show the existence of "nice" subsources $\tilde{X}, \tilde{Y}$ of the original sources. In the second step we used Claims 8.5 and Claim 8.6 to show that on "nice" sources $\tilde{X}, \tilde{Y}$ the properties of Claim 8.3 follow. To prove the stronger version we replace the Claims in the first step with stronger versions that show the existence of a large subsource collection of nice sources. The second step then remains unchanged. We have already observed that the proof of Claim 8.8 gives a stronger version that shows the existence of a large subsource collection (this is stated precisely in Claim 8.10). Thus, we only need to state and prove a stronger version of Claim 8.7. A stronger version which shows the existence of a large subsource collection is stated next.

Claim 9.2. *There exists a constant $\mu > 0$ (that depends only on δ and m) such that the following holds: For every X, Y independent random variables over $\{0,1\}^n$ with $H^\infty(X), H^\infty(Y) > \delta n$ there exists a partition $\tilde{I}$, an integer w and a subsource collection $X'_1, \ldots, X'_w$ and $Y'_1, \ldots, Y'_w$ of (X, Y) of measure μ such that for any $j \in [w]$ when partitioning X'_j, Y'_j according to $\tilde{I}$ (which we denote by $(X'_j)_1, (X'_j)_2, (X'_j)_3$ and $(Y'_j)_1, (Y'_j)_2, (Y'_j)_3$ then:*

- $H^\infty((X'_j)_1), H^\infty((Y'_j)_1) \geq \delta n/2$.
- $H^\infty((X'_j)_2), H^\infty((Y'_j)_2) \geq \delta^2 n/50$.
- $H^\infty((X'_j)_3), H^\infty((Y'_j)_3) = 0$ *(or in other words that both $(X'_j)_3, (Y'_j)_3$ are fixed constants).*

We prove Claim 9.2 in Section 9.2. We are now ready to prove Claim 9.1. As explained above the proof is essentially identical to that of Claim 8.3.

Proof of Claim 9.1 Let X, Y be independent δ-sources. By the combination of Claim 9.2 and Claim 8.10 we get using Fact 4.5 that there exists a partition $\tilde{I}$ an integer w and a subsource collection $\tilde{X}_1, \ldots, \tilde{X}_w$ and $\tilde{Y}_1, \ldots, \tilde{Y}_w$ of (X, Y) that has a measure which is larger than some positive constant such that for every $j \in [w]$ the subsources $\tilde{X}_j$ of X and $\tilde{Y}_j$ of Y are nicely structured according to $\tilde{I}$ and furthermore, $\Pr[\texttt{test}_{\tilde{I}}(\tilde{X}_j, \tilde{Y}_j) = 1] = 1$. The first item of Claim 9.1 follows directly from Claim 8.5. For the second item we note that by Claim 8.6 for every $j \in [w]$:

$$\Pr[\exists I \not\succeq \tilde{I} : \ \texttt{test}_I(\tilde{X}_j, \tilde{Y}_j) = 1] \leq 2^{-10m}$$

Whenever the event above does not hold we have that $\tilde{I}$ is the partition that is selected and thus, $\tilde{I}$ is selected with probability at least $1 - 2^{-10m}$ as required.

9.2 Proof of Claim 9.2

We consider each of the two sources separately. We prove the following Claim.

Claim 9.3. *Let X be a δ-source then there exists $i_1 \in [t]$ and a subsource X' of X with measure $1/4t$ such that when partitioning according to i_1 for every string x_3 of length $(t - i_1)\delta n/10$*

- $H^\infty(X'_1 | X'_3 = x_3) \geq \delta n/2$.
- $H^\infty(X'_2 | X'_3 = x_3) \geq \delta^2 n/50$.

Once we prove this claim we can apply it on both sources to get subsources X' of X and Y' of Y and indices (i_1, i_2). We define the partition $\tilde{I} = (i_1, i_2)$. For any choice (x_3, y_3) of values of X_3, Y_3 that occur with positive probability under (X', Y') we define $X'_{x_3,y_3} = (X' | X'_3 = x_3)$ and $Y'_{x_3,y_3} = (Y' | Y'_3 = y_3)$. Note that the collection $(X'_{x_3,y_3})_{(x_3,y_3)}$ and $(Y'_{x_3,y_3})_{(x_3,y_3)}$ are a subsource collection of (X, Y) of measure $1/16t^2$ with the required properties.

We are left with proving Claim 9.3. For every $i \in [t]$ and x in the support of X we consider partitioning according to i and define $r_i(x) = \Pr[X_2 = x_2 | X_3 = x_3]$. Note that for any x in the support of X:

$$2^{-\delta n} \geq \Pr[X = x] = \prod_{i \in [t]} r_i(x)$$

This is because when varying i from t to 1 each new term is the conditional probability that $X_2^i = x_2^i$ conditioned on the event that the blocks in X_3^i are fixed.

For any string x there exists an i such that $r_i(x) \leq 2^{-\frac{\delta n}{4t}}$ (as otherwise the product cannot be smaller than $2^{-\delta n}$). For every x let $i(x)$ be the largest index i with that property. For every $i \in [t]$ let $B_i = \{x : \Pr[X = x] > 0 \text{ and } i(x) = i\}$. Note that the sets $\{B_i\}_{i\in[t]}$ are a partition of the support of X and therefore there exists an i_1 such that $\Pr[B_{i_1}] \geq 1/t$. From now on we always partition according to i_1 and to simplify the notation we define $B = B_{i_1}$. Let $B' = \{x \in B : \Pr[X \in B | X_3 = x_3] < 1/2t\}$. Note that

$$\begin{aligned}
\Pr[X \in B'] &= \textstyle\sum_{x_3} \Pr[X \in B' | X_3 = x_3] \Pr[X_3 = x_3] \\
&\leq \textstyle\sum_{x_3} \Pr[X \in B | X_3 = x_3] \Pr[X_3 = x_3] \\
&\leq \tfrac{1}{2t} \cdot \textstyle\sum_{x_3} \Pr[X_3 = x_3] \\
&\leq \tfrac{1}{2t}
\end{aligned}$$

Let $A = B \setminus B'$ it follows that $\Pr[X \in A] \geq 1/2t$. Let $X' = (X | X \in A)$. We have that X' is a subsource of X with measure $1/2t$. It remains to show that X' fulfils the required properties.

Fix some string $x \in A$ (that is in the support of X') we estimate $H^\infty(X'_2 | X'_3 = x_3)$ and $H^\infty(X'_1 | X'_3 = x_3)$.

$$\begin{aligned}
\Pr[X'_2 = x_2 | X'_3 = x_3] &= \Pr[X_2 = x_2 | X_3 = x_3 \wedge X \in A] \\
&= \tfrac{\Pr[X_2 = x_2 \wedge X \in A | X_3 = x_3]}{\Pr[X \in A | X_3 = x_3]} \\
&\leq \tfrac{\Pr[X_2 = x_2 | X_3 = x_3]}{1/2t} \\
&\leq 2t \cdot 2^{-\frac{\delta n}{4t}} \\
&\leq 2^{-\delta^2 n/40 + \log t + 1} \\
&\leq 2^{-\delta^2 n/50}
\end{aligned}$$

It follows that $H^\infty(X'_2 | X'_3 = x_3) \geq \delta^2 n/50$ as required. We now estimate $H^\infty((X'_1, X'_2) | X'_3 = x_3)$. The argument is very similar to the previous one. We observe that for any $x \in B$ (and therefore for any $x \in A$) we have that

$$\Pr[X_3 = x_3] > 2^{-\frac{(t-i)\delta n}{4t}} > 2^{-\delta n/4}$$

This is because for any $x \in B$, i_1 is the largest index such that $r_i(x) \leq 2^{-\frac{\delta n}{4t}}$. Note that $2^{-\delta n} \geq \Pr[X = x] = \Pr[(X_1, X_2) = (x_1, x_2) | X_3 = x_3] \cdot \Pr[X_3 = x_3]$ and therefore for any $x \in B$, $\Pr[(X_1, X_2) = (x_1, x_2) | X_3 = x_3] \leq 2^{-(\delta n - \delta n/4)} \leq 2^{-\frac{3\delta n}{4}}$. Using this inequality we can repeat the argument above.

$$
\begin{aligned}
\Pr[X_1' = x_1 \wedge X_2' = x_2 | X_3' = x_3] &= \Pr[X_1 = x_1 \wedge X_2 = x_2 | X_3 = x_3 \wedge X \in A] \\
&= \frac{\Pr[X_1 = x_2 \wedge X_2 = x_2 \wedge X \in A | X_3 = x_3]}{\Pr[X \in A | X_3 = x_3]} \\
&\leq \frac{\Pr[X_1 = x_1 \wedge X_2 = x_2 | X_3 = x_3]}{1/2t} \\
&\leq 2t \cdot 2^{-\frac{3\delta n}{4}} \\
&\leq 2^{-3\delta n/4 + \log t + 1} \\
&\leq 2^{-2\delta n/3}
\end{aligned}
$$

It follows that $H^\infty((X_1', X_2')|X_3' = x_3) \geq 2\delta n/3$. Observe that by Lemma 3.2

$$
H^\infty(X_1'|X_3' = x_3) \geq H^\infty((X_1', X_2')|X_3' = x_3) - |X_2'| \geq \frac{2\delta n}{3} - \frac{\delta n}{10} \geq \frac{\delta n}{2}
$$

as required. This concludes the proof of the Claim.

10 Conclusion and open problems

The two main results of this paper are new constructions of 3-source extractors and 2-source dispersers for every min-entropy rate $\delta > 0$. While these constructions significantly improve the parameters achieved by the best previous explicit constructions they are still far from the parameters achieved by "non-explicit" objects which are shown to exist using the probabilistic method.

The construction and analysis of the 2-source disperser are quite complicated and it will be very interesting to give a simpler construction.

Our constructions use brute force search to find a constant sized optimal extractor. We remark that in a subsequent work Rao [Rao06] constructs an extractor for two "block-wise sources" and that this construction does not use brute force search. Such extractors extract randomness from two independent nicely structured sources and can therefore replace the function $\texttt{disp}_I$ which is a component in the construction of the Disperser (and give a construction that does not rely on brute force search).

We remark that in a subsequent work, Barak et al [BRSW06] construct an improved 2-source disperser. This construction heavily builds on this paper and in particular relies on a (recursive and more complicated) implementation of the "challenge-response mechanism" presented in this paper.

Acknowledgements

We are grateful to Amnon Ta-Shma for helpful comments that significantly improved the presentation.

References

[BIW04] B. Barak, R. Impagliazzo, and A. Wigderson. Extracting Randomness Using Few Independent Sources. *SIAM J. Comput*, 36(4):1095–1118, 2006. Preliminary version in FOCS '04.

[BRSW06] B. Barak, A. Rao, R. Shaltiel, and A. Wigderson. 2-source dispersers for sub-polynomial entropy and Ramsey graphs beating the Frankl-Wilson construction. In *Proc. of the 38th Annual ACM Symposium on Theory of Computing*, pages 671–680, 2006.

[BST03] B. Barak, R. Shaltiel, and E. Tromer. True Random Number Generators Secure in a Changing Environment. In *Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, pages 166–180, 2003. LNCS no. 2779.

[Blu86] M. Blum. Independent Unbiased Coin Flips from a Correlated Biased Source–A Finite State Markov Chain. *Combinatorica*, 6(2):97–108, 1986.

[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. *Geom. Funct. Anal.*, 14(1):27–57, 2004.

[CRVW02] M. Capalbo, O. Reingold, S. Vadhan, and A. Wigderson. Randomness conductors and constant-degree lossless expanders. In *Proc. 34th STOC*, pages 659–668. ACM, 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.

[CGH+85] B. Chor, O. Goldreich, J. Hastad, J. Friedman, S. Rudich, and R. Smolensky. The Bit Extraction Problem of t-Resilient Functions (Preliminary Version). In *Proc. 26th FOCS*, pages 396–407. IEEE, 1985.

[CW89] A. Cohen and A. Wigderson. Dispersers, Deterministic Amplification, and Weak Random Sources. In *Proc. 30th FOCS*, pages 14–19. IEEE, 1989.

[DEOR04] Y. Dodis, A. Elbaz, R. Oliveira, and R. Raz. Improved Randomness Extraction from Two Independent Sources. In *Proc. of 8th RANDOM*, 2004.

[DS02] Y. Dodis and J. Spencer. On the (non)Universality of the One-Time Pad. In *Proc. 43rd FOCS*, pages 376–388. IEEE, 2002.

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

[GRS04] A. Gabizon, R. Raz, and R. Shaltiel. Deterministic Extractors for Bit-Fixing Sources by Obtaining an Independent Seed. *SIAM J. Comput.*, 36(4):1072–1094, 2006. Preliminary version in FOCS '04.

[GS08] A. Gabizon and R. Shaltiel. Increasing the Output Length of Zero-Error Dispersers. In *APPROX-RANDOM*, volume 5171 of *Lecture Notes in Computer Science*, pages 430–443. Springer, 2008.

[Gol95] O. Goldreich. Three XOR-Lemmas – An Exposition. ECCC Report TR95-056, 1995.

[GUV09] V. Guruswami, C. Umans, and S. Vadhan. Unbalanced expanders and randomness extractors from Parvaresh–Vardy codes. *Journal of the ACM*, 56(4):20, 2009. Preliminary version in CCC '07.

[KZ06] J. Kamp and D. Zuckerman. Deterministic Extractors for Bit-Fixing Sources and Exposure-Resilient Cryptography. *SIAM J. Comput.*, 36(5):1231–1247, 2007. Preliminary version in STOC '06.

[KS09] N. Katz and C. Shen. Garaev's inequality in finite fields not of prime order. Preprint, Arxiv:math/0703676, 2009.

[LRVW03] C.-J. Lu, O. Reingold, S. Vadhan, and A. Wigderson. Extractors: optimal up to constant factors. In *Proc. 35th STOC*, pages 602–611. ACM, 2003.

[MP90] J. L. McInnes and B. Pinkas. On the Impossibility of Private Key Cryptography with Weakly Random Keys. In *Crypto '90*, pages 421–436, 1990. LNCS No. 537.

[MU01] E. Mossel and C. Umans. On the complexity of approximating the VC dimension. *J. Comput. Syst. Sci.*, 65(4):660–671, 2002. Preliminary version in CCC '01.

[OL89] B. Or and Linial. Collective Coin Flipping. *ADVCR: Advances in Computing Research*, 5, 1989.

[Pud06] P. Pudlak. On Explicit Ramsey Graphs and Estimates on the Numbers of Sums And Products. In *Topics in discrete mathematics, Algorithms Combin.*, volume 26, pages 169–175. Springer, 2006.

[PR04] P. Pudlák and V. Rödl. Pseudorandom sets and explicit constructions of Ramsey graphs. In *Complexity of computations and proofs*, pages 327–346. Quad. Mat., 13, Dept. Math., Seconda Univ. Napoli, Caserta, 2004.

[Rao06] A. Rao. Extractors for a constant number of polynomially small min-entropy independent sources. In *Proc. of the 38th Annual ACM Symposium on Theory of Computing*, pages 497–506, 2006.

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

[RR99] R. Raz and O. Reingold. On Recycling the Randomness of States in Space Bounded Computation. In *Proc. of the 31st Annual ACM Symposium on Theory of Computing*, pages 159–168, 1999.

[RSW00] O. Reingold, R. Shaltiel, and A. Wigderson. Extracting Randomness via Repeated Condensing. *SIAM J. Comput.*, 35(5):1185–1209, 2006. Preliminary version in FOCS '00.

[RVW00] O. Reingold, S. P. Vadhan, and A. Wigderson. Entropy waves, the zig-zag graph product, and new constant-degree expanders. *Annals of Mathematics*, 155(1):157–187, 2002. Preliminary vesion in FOCS '00.

[SV84] M. Santha and U. V. Vazirani. Generating Quasi-Random Sequences from Slightly-Random Sources. In *Proc. 25th FOCS*, pages 434–440. IEEE, 1984.

[Sha02] R. Shaltiel. Recent developments in extractors. *Bulletin of the European Association for Theoretical Computer Science*, 2002.

[TS96] A. Ta-Shma. On Extracting Randomness From Weak Random Sources. In *STOC*, pages 276–285, 1996.

[TSUZ07] A. Ta-Shma, C. Umans, and D. Zuckerman. Lossless Condensers, Unbalanced Expanders, And Extractors. *Combinatorica*, 27(2):213–240, 2007.

[TV06] T. Tao and V. Vu. *Additive combinatorics*. Cambridge University Press, 2006.

[TV00] L. Trevisan and S. Vadhan. Extracting randomness from samplable distributions. In *Proc. 41st FOCS*, pages 32–42. IEEE, 2000.

[Vaz87] U. Vazirani. Strong Communication Complexity or Generating Quasi-Random Sequences from Two Communicating Semi-Random Sources. *Combinatorica*, 7(4), 1987.

[vN51] J. von Neumann. Various Techniques Used in Connection with Random Digits. *Applied Math Series*, 12:36–38, 1951.

[Zuc90] D. Zuckerman. General Weak Random Sources. In *Proc. 31st FOCS*, pages 534–543. IEEE, 1990.

[Zuc06] D. Zuckerman. Linear Degree Extractors and the Inapproximability of Max Clique and Chromatic Number. *Theory of Computing*, 3(1):103–128, 2007. Preliminary version in STOC '06.

A Proof of Lemma 7.1

For completeness we give the proof of Lemma 7.1. This is a standard calculation that follows from the following more general lemma.

Lemma A.1. *Let $m < k < d$ be integers, and let $\epsilon > 0$. There exist some constant $a > 0$ such that if $k > \log d + 2m + 2\log(1/\epsilon) + a$ then there exists a $2^{5d^2 \cdot 2^{2k}}$-time computable 2-source extractor $\mathtt{opt} : \{0,1\}^{d\times 2} \to \{0,1\}^m$ with k-entropy requirement and distance ϵ.*

Proof. We first give a probabilistic argument that such an extractor exists, and then find it by brute force. Let $M = 2^m$, $D = 2^d$ and $K = 2^k$. Let A be a randomly chosen $D \times D$ matrix with entries in $\{0,1\}^m$, where the $D^2 m$ bits used as an entries of A are $K^2 m$-wise independent. We show that with positive probability the function $\mathtt{opt}(x,y) = A_{x,y}$ is an extractor with the required parameters. We say that R is a *rectangle* if $R \in \binom{[D]}{K} \times \binom{[D]}{K}$. We first show that for every rectangle R, nonzero $v \in \{0,1\}^m$ and $\epsilon > 0$

$$\Pr\left[\Big|\sum_{(i,j)\in R} (-1)^{<A_{ij},v>}\Big| > K^2\epsilon/M\right] < e^{-\Omega(\frac{\epsilon^2 K^2}{M^2})}. \tag{1}$$

For every $(i,j) \in R$ and nonzero $v \in \{0,1\}^m$ define a random variable $B_{ij} = (-1)^{<A_{ij},v>}$ and let $B = \sum_{(i,j)\in R} B_{ij}$. Note that $\mathbb{E}[B] = 0$. Moreover, since every K^2 entries of A are independent, we have the independence of variables B_{ij}. Therefore the Chernoff bound implies that $\Pr\left[|B| > \epsilon K^2/M\right] < e^{-\Omega(\frac{\epsilon^2 K^2}{M^2})}$.

We say that A passes R and v if $\left|\sum_{(i,j)\in R}(-1)^{<A_{ij},v>}\right| \le K^2\epsilon/M$. By a union bound over all rectangles and nonzero vectors we get that the probability that there exist a rectangle R and nonzero v such that A does not pass R and v is bounded by

$$\binom{D}{K}^2 \cdot M \cdot e^{-\Omega(\frac{\epsilon^2 K^2}{M^2})} \le 2^{3Kd - \Omega(\frac{\epsilon^2 K^2}{M^2})}$$

It is easy to verify that there exists some constant $c > 0$ such that the right side of the above inequality is smaller than one when $K \geq \frac{cdM^2}{\epsilon^2}$. This holds for $k > \log d + 2m + 2\log(1/\epsilon) + O(1)$. Thus, there exists a matrix A' which passes all rectangles and all vectors v. Let $\texttt{opt}(x, y) = A'_{x,y}$. We now verify that $\texttt{opt}$ is a 2-sample extractor with k-entropy requirement and distance ϵ. It is sufficient consider only pairs of independent sources where each component is a flat distribution. Each such pair of sources is uniformly distributed over some rectangle R. Fix such an R and let Y denote the output distribution of $\texttt{opt}$ on R. For every nonzero $v \in \{0,1\}^m$ we have that the expected bias of Y in v is $\left|\mathbb{E}[(-1)^{<Y,v>}]\right| \leq \epsilon/M$. Thus, the distribution Y is ϵ/M-biased and by the Vazirani XOR-lemma (see e.g. [Gol95]) we have that Y is ϵ-close to uniform. (In fact, it also would have followed if Y were only $\epsilon/\sqrt{M}$-biased). This shows that $\texttt{opt}$ indeed is an extractor.

To find a matrix which passes all rectangles and vectors we go over all possible matrices A in the appropriate sample space. There are efficient constructions of such a space that are of size $(D^2m)^{K^2m}$. For each matrix we check all rectangles and vectors, altogether $\binom{D}{K}^2 \cdot M$ choices. Finally for each such choice we check whether (1) holds in time K^2m. Thus, the total time we need is bounded by

$$(D^2m)^{K^2m} \cdot \binom{D}{K}^2 \cdot M \cdot K^2m \leq 2^{3d^2 \cdot 2^{2k} + d \cdot 2^{k+1} + m + 3k} \leq 2^{5d^2 \cdot 2^{2k}}$$

□

ECCC ISSN 1433-8092
http://eccc.hpi-web.de