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

Boaz Barak*
Institute for Advanced Study
Princeton, NJ
boaz@ias.edu

Guy Kindler†
Institute for Advanced Study
Princeton, NJ
gkindler@ias.edu

Ronen Shaltiel‡
Dept. Comp. Sci.
Haifa University, ISRAEL
ronen@cs.haifa.ac.il

Benny Sudakov§
Dept. of Mathematics
Princeton University
bsudakov@math.princeton.edu

Avi Wigderson¶
Institute for Advanced Study
Princeton, NJ
avi@ias.edu

ABSTRACT

A distribution X over binary strings of length n has min-entropy k if every string has probability at most 2^{-k} in X.[1] We say that X is a δ-source if its rate k/n is at least δ.

We give the following new explicit constructions (namely, poly(n)-time computable functions) of *deterministic* extractors, dispersers and related objects. All work for any fixed rate $\delta > 0$. No previous explicit construction was known for either of these, for any $\delta < 1/2$. The first two constitute major progress to very long-standing open problems.

1. **Bipartite Ramsey** $f_1 : (\{0,1\}^n)^2 \to \{0,1\}$, such that for any two independent δ-sources X_1, X_2 we have $f_1(X_1, X_2) = \{0,1\}$. This implies a new explicit construction of $2N$-vertex bipartite graphs where no induced N^δ by N^δ subgraph is complete or empty.

2. **Multiple source extraction** $f_2 : (\{0,1\}^n)^3 \to \{0,1\}$, such that for any three independent δ-sources X_1, X_2, X_3 we have that $f_2(X_1, X_2, X_3)$ is ($o(1)$-close to being) an unbiased random bit.

3. **Constant seed condenser**[2] $f_3 : \{0,1\}^n \to (\{0,1\}^m)^c$, such that for any δ-source X, one of the c output distributions $f_3(X)_i$, is a 0.9-source over $\{0,1\}^m$. Here c is a constant depending only on δ.

4. **Subspace Ramsey** $f_4 : \{0,1\}^n \to \{0,1\}$, such that for any *affine*-δ-source[3] X we have $f_4(X) = \{0,1\}$.

The constructions are quite involved and use as building blocks other new and known gadgets. But we can point out two important themes which recur in these constructions. One is that gadgets which were designed to work with independent inputs, sometimes perform well enough with correlated, high entropy inputs. The second is using the input to (introspectively) find high entropy regions within itself.

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

General Terms: Theory.

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

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.

*Supported by the state of New Jersey and by NSF grant CCR-0324906.

†Supported by NSF grants DMS-0111298 and CCR-0324906.

‡Part of this research was done while staying at the Weizmann Institute and supported by the Koshland scholarship

§Supported by NSF grant DMS 0355497 and an Alfred P. Sloan Foundation Fellowship

¶Supported by NSF grant CCR-0324906.

STOC'05, May 22-24, 2005, Hunt Valley, Maryland, USA.

[1] It is no real loss of generality, and very convenient, to think of X as a uniform distribution on some set of 2^k elements (in which case min-entropy is the same as Shannon entropy).

[2] This result was also independently obtained by Ran Raz [22].

[3] This is a uniform distribution over an affine subspace of dimension at least δn.

Much of this research assumes only that the given "weak source" has sufficient (min)-entropy, and that extractors can use an extra, short random "seed" to aid in distilling the randomness. Such a seed (of logarithmic length) is easily seen to be necessary. This research focused on explicitly constructing extractors of small seed (and long output). The survey [25] explains most of the recent developments, and the paper [17] contains the current state-of-art construction 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. [18, 13, 2]). To do without it, one must impose extra conditions (beyond entropy content) on the class of sources from which one extracts. This body of work, which includes [29, 5, 24, 9, 3, 28, 11, 10, 27, 19, 16, 15] considers a variety of restrictions, mostly structural, on the given sources, and shows how to extract from them deterministically (without seed).

This paper provides several new explicit constructions of seedless extractors, dispersers and related objects (all to be defined), greatly 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. For the sake of brevity and clarity, we would 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 [24], and further studied by Chor and Goldreich [9] (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 if for every ℓ sources $X_1, \cdots, X_\ell$, each with min-entropy k, the distribution $f(X_1, \cdots, X_\ell)$ (obtained by applying f on an independent sample from each source) is close to uniform on $\{0,1\}^m$.

For this model, the probabilistic method easily gives the best that can be hoped for. Two sources and $\log n$ 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 (poly(n)-time), it will come as a building block in our constructions). We call such a function $\texttt{opt}$ (for Optimal Extractor).

For two sources, the best explicit construction requires in contrast min-entropy $> n/2$. 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 [9] to be such an extractor. Moreover, a natural variant $\texttt{Had}'$ which outputs $m = \Omega(n)$ was shown to be an extractor in [28] under the same conditions (see a simplified and improved analysis in [12]). To date, no one has found an explicit 2-source extractor with sources of rate 1/2 or less[4]. Indeed, until last year no extractor breaking the 1/2 rate barrier was known using less than $\ell = \text{poly}(n)$ number of sources. Still, the Hadamard function and its extension to long outputs $\texttt{Had}'$ will be an important building block in our construction.

Last year [1] gave a extractor that, for any $\delta > 0$ uses only a constant $\ell = \text{poly}(1/\delta)$ δ-sources.[5] 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*[6] *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 can also increase the output length m to be a constant fraction of the input entropy using 7 independent sources. In both constructions the error (=distance of the output to the uniform distribution) is only sub-constant (about $1/\log\log n$), and as yet we have no idea how to make it exponentially, or even polynomially small. Subsequently to us, Raz [22] (see also Section 1.5) used our construction (as well as additional ideas) to obtain a 3-source extractor which outputs $\Omega(n)$ number of bits, and works with one δ-source and two sources of logarithmic entropy.

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 [14] only achieves a much larger value: $K = 2^{\Theta(\sqrt{\log N \log\log N})}$.

Bipartite Ramsey graphs. An even harder variant of this problem is the *bipartite Ramsey problem:*

[4]We note that using Weil's estimates the Paley matrix (i.e., for prime p the function $P : \mathbb{F}_p^2 \to \{0,1\}$ where $P(x, y) = 1$ iff $x - y \pmod p$ is a quadratic residue) can be proven to be an extractor even if one source has only about $\log n$ entropy. However the other source still has to have entropy $> n/2$

[5]This extractor was proposed already in [30], but the analysis there relies on an unproven number theoretic assumption.

[6]This function, and all the other functions we construct, is computable by a deterministic polynomial-time (uniform) Turing machine.

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 $K = N^\delta$ (which is in turn a coloring for the non-bipartite version). Until recently, the best known explicit 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 [21].[7]

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 $\texttt{disp}(X_1, X_2) = \{0,1\}^m$. 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)[8]. 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 with any constant (and even slightly sub-constant) rate $\delta > 0$ (so any $K > N^\delta$ in the bipartite Ramsey problem) and any *constant* output length m. Our disperser is strong in the sense that it obtains every output 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. That is, we prove the following theorem:

Theorem 1.2 (Two-source disperser). *For every constants $\delta > 0$ and $m \in \mathbb{N}$, 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 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 $c = c(m, \delta)$, such that for every $z \in \{0,1\}^m$, $\Pr[\texttt{disp}(X_1, X_2) = z] \geq c(m, \delta)$.*

1.3 Constant seed (1-source) 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. Both in the Mathematical sense that an extractor is an ultimate condenser (its output has maximum possible rate), as well as in the practical sense – some constructions of extractors proceed by iterated condensing. Various condensers appear in [23, 26, 17, 8] and other works, mainly as building blocks to constructing extractors and expanders.[9]

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.[10]

New Results. We give the first explicit constant seed condenser for linear entropy. More precisely, for every $\delta > 0$ there are integers c, d and a $\mathrm{poly}(n)$-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 $\texttt{con}(X)_i$ (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, but all we shall need is any constant $> 1/2$). The constant seed may be viewed as selecting the output block at random. In the paper we prefer the view of outputting several blocks, and call this construction a *somewhere condenser*. The theorem that we prove is the following:

Theorem 1.3 (Somewhere condenser). *For every constant $\delta > 0$, there exists constants $\epsilon > 0$ and $\ell \in \mathbb{N}$ and a $\mathrm{poly}(n)$-time computable function $\texttt{con} : \{0,1\}^m \to \{0,1\}^{(\epsilon n) \times \ell}$ such that for every δ-source X over $\{0,1\}^n$ there exists a random variable I (correlated with X) over $[\ell]$ such that $\texttt{con}(X)_I$ is within $2^{-\epsilon n}$ statistical distance to the uniform distribution over $\{0,1\}^{\epsilon n}$, where $\texttt{con}(X)_I$ denotes the I^{th}-block of $\texttt{con}(X)$.*

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.

The condenser above is obtained by iterating a constant number of times a *basic* condenser $\texttt{bcon}$ (described in the Section 2.1), which uses only a 2-bit seed (namely has 4 output blocks) and increases the entropy rate by a constant amount.

We note that, independently from us, Ran Raz [22] has obtained a different constant seed condenser construction, which also use the [1] paper, but in another way. (In fact, using a new variant of the [17] merger, Raz constructs a condenser with the advantage that *most* of the output blocks are condensed.) We elaborate on this and also some other subsequent works in Section 1.5.

[7] 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 $n = \log N$.

[8] 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 $\{0,1\}^m$. We use the 0-error definition, which is more relevant to our setting.

[9] We note that in some of these works the definition is relaxed to allow situations when the output rate is *smaller* than the input rate, as long the the output length is shorter - these gadgets turn out to be useful as well.

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

1.4 A disperser for affine subspaces

Background. There are some other natural restrictions on sources (beyond allowing a few independent ones) that admit deterministic extraction. Two such models are bit-fixing sources [10, 3, 11, 19, 16, 15] and efficiently samplable distributions [27]. In the bit-fixing model, an (unknown) subset of δn bits (out of the n input bits) is uniformly distributed, and the rest of the bits are fixed to some unknown value.[11] In the "efficiently samplable distributions" model the source is obtained by applying an *efficiently computable* "sampling" function $g : \{0,1\}^{\delta n} \to \{0,1\}^n$ on a uniformly distributed input. Trevisan and Vadhan [27] give a construction (based on unproven complexity theoretic assumptions) that works for sources with entropy rate $\delta > 1/2$ that are sampled by polynomial sized circuits g. (In fact, in that construction $\delta \geq (1-\gamma)$ for some small constant γ).

In this paper we consider sources which are uniformly distributed over an affine subspace of $\mathrm{GF}(2)^n$. Such sources can be seen as a natural generalization of bit-fixing sources (since any bit-fixing source is in particular a distribution over an affine subspace), or alternatively, as a restriction of the model of [27] in which the sampling function g is affine (note that every such function can be computed in size n^2).

Strangely, what was known about this model is very similar quantitatively to the two independent source model in the sense one one hand there is a (nonconstructive) "optimal extractor" for entropy $\Theta(\log n)$ but on the other hand an explicit extractor was known only for entropy rate $\delta > 1/2$. (In fact, this explicit extractor is no other than the Hadamard function `Had` mentioned above, applied to two halves of the sample [4].)

New Results. We give an explicit zero-error disperser for affine sources of entropy rate δ, that outputs constantly many bits. As in the case of the two-source disperser, the affine-source disperser is strong, obtaining each output string with at least a constant probability. That is, we prove the following result

Theorem 1.4 (Seedless affine disperser). *Let $\delta > 0$ and $m \in \mathbb{N}$ be some constants. Then, there exists a $\mathrm{poly}(n)$-time computable function $\texttt{a-disp} : \{0,1\}^n \to \{0,1\}^m$ such that for every affine-subspace $X \subseteq \{0,1\}^m$ with dimension at least δn, $\texttt{a-disp}(X) = \{0,1\}^m$. Moreover, there exists a constant $d = d(m,\delta)$, such that for every $z \in \{0,1\}^m$, $\Pr_{x\in_R X}[\texttt{a-disp}(x) = z] \geq d(m,\delta)$.*

For 1-bit output a Ramsey theoretic interpretation of the result suggests itself – we give a 2-coloring of the Boolean cube F_2^n, in which no affine subspace of linear dimension is monochromatic. Due to space considerations, we defer all details of this construction to the appendix. It proceeds by first showing a somewhere extractor for affine subspaces (which uses the [1] extractor on different parts of the source). We then use this somewhere affine-extractor to obtain an affine-disperser in a way very similar to the way we use our 2-source somewhere extractor to obtain a 2-source disperser.

[11] In the terminology of the above papers this is the so-called *oblivious* bit-fixing model.

Note that the new results here are quantitatively the same as our 2-source results. As mentioned above, the techniques are related as well, but at this point this fact may be surprising - there seem to be little resemblance between the models, and indeed there seem to be no reductions between them in either direction. The similarity in techniques may simply be a byproduct of the fact that we were working on them in parallel, and progress on one suggested related progress on the other (note that at some point we do use the two-source Ramsey construction as a black box in the affine-source disperser, but we could have just as well used a direct construction there). This simply manifests the generality of our techniques, however it would be interesting to find any tighter connections between the two models.

1.5 Some independent and subsequent works.

As mentioned earlier, Ran Raz [22] has some results related to ours, parts of which are independent of ours and parts of which are subsequent to ours. Independently of this work, Raz gives an alternative construction of a somewhere condenser (that also builds on the construction of [1]). He also shows how to obtain a condenser where *most* of the blocks are condensed (as opposed to only one of them, as in our case). He also shows how to generalize some of our results to the *asymmetric* case, where each source can have different length and entropy. We also note that subsequently to this work, Zuckerman (Personal Communication, 2005) gave an somewhat improved version of our condenser, with a simplified analysis, which is based on another result in [7] (implied by the sum-product theorem): an estimate on the number of incidences of points on lines in prime fields.

Subsequently to this work, Pudlak [20] gave a different (and simpler) construction of a coloring of the full bipartite N by N graph with 3 colors such that no subset of size $N^{1/2-\epsilon}$ by $N^{1/2-\epsilon}$ is monochromatic for some fixed small ϵ. In a very recent work, Bourgain [6] gave a statistical version of Pudlak's construction, namely an *extractor* for two sources of entropy rate $1/2 - \epsilon$ for some small constant $\epsilon > 0$. This extractor outputs $\Omega(n)$ bits with statistical distance $2^{-\Omega(n)}$ (where $n = \log N$). Both these works also use the above mentioned points vs. lines estimate.

2. TECHNIQUES AND OVERVIEW OF THE MAIN CONSTRUCTIONS

Unfortunately, due to space consideration we have no room for the proofs of our results (except for the condenser) in this extended abstract. However, in the following subsections we describe some detail some of the extra gadgets we develop, and how to compose them with each other and with known gadgets to get the constructions above. By far the most difficult is the 2-source disperser, and the ideas leading to it take several subsections to describe even partially. Still, reading these will simplify understanding the formal proofs (which can be found in the full version of this paper).

2.1 A 2-bit seed condenser

Here we describe our basic condenser bcon, which uses only a 2-bit seed to increase the entropy rate of any source by a constant amount. Iterating it a constant number of times on a δ-source allows us to increase to rate (of some output block) above 0.9 (say), thus giving the condenser con described in Section 1.3.

Our basic condenser bcon will take strings of length n with $n = 3p$ for some prime p.[12] For every $x \in \{0,1\}^n$ let $x = x_1x_2x_3$ its natural partition to three length p blocks. Define $\texttt{bcon} : \{0,1\}^{3p} \to (\{0,1\}^p)^4$ by $\texttt{bcon}(x) = x_1, x_2, x_3, x_1 \cdot x_2 + x_3$ (with arithmetic in $GF(2^p)$).

We prove that in X is a δ-source with $\delta < 0.9$, then at least one of the output blocks is a $(\delta + \Omega(\delta^2))$-source. Using this, we prove the following theorem:

Theorem 2.1 (Basic condenser). *There exists a constant $\alpha > 0$ and a polynomial-time computable somewhere condenser* $\texttt{bcon} : \{0,1\}^n \to \{0,1\}^{(n/3)\times 4}$ *that satisfies the following: For every δ-source X (with $\delta \gg n^{-1/4}$), there exists a dependent random variable I over* [4] *such that $\texttt{bcon}(X)_I$ is within statistical distance $\epsilon = 2^{-\alpha\delta^2 n}$ to having rate at least $(1 + \alpha\delta)\delta$.*

The proof (which is in Section 3) heavily relies on the main lemma of [1], who proved $x_1 \cdot x_2 + x_3$ is condensed *assuming* that the x_i's are *independent*. We certainly *cannot* assume that in our case, as X is a general source. Still, we use that lemma to show that if none of these first 3 blocks is more condensed than the input source, then they are "independent enough" for using that main lemma.

2.2 A 2-source constant-seed/"somewhere" extractor

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

What we prove is that for every $\delta > 0$ there are integers c, d and a poly(n)-time computable function $\texttt{s\_ext} : (\{0,1\}^n)^2 \to (\{0,1\}^{n/c})^d$, such that for every two 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 Vazirani variant Had′ of the Hadamard 2-source extractor for rate $> 1/2$ [28] 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 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.

[12] Note that if n is not of this form than it can be converted to this form by padding the input with $o(n)$ additional bits using standard number theoretic bounds on the density of primes.

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 nonuniform optimal 2-source extractor opt.

Recall that our 2-source *somewhere* extractor s_ext produces a constant number (say) d of linear length output blocks, one of which is random. First we note that producing shorter output blocks maintains this property[13].

Let us indeed output only a constant b bits in every block (satisfying $b \geq \log(db)$). Concatenating all output blocks of this $\texttt{s\_ext}(X_1, X_2)$ gives us a distribution (say Z_1) on db 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 (non-uniform) 2-source extractors on Z_1, Z_2 to get a uniform bit; as db is only a constant, such an extractor can be found 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 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. Fixing a random sample x_2 of X_2 will guarantee the output properties of both s_ext's with high probability *and* keep the two inputs to opt independent (since once x_2 is fixed the distributions $\texttt{s\_ext}(X_1, x_2)$ and $\texttt{s\_ext}(X_3, x_2)$ are independent).

2.4 A better 2-source somewhere extractor

Our somewhere extractor s_ext produced a constant number of output blocks, one of which is random. It gave us a 4-source (and even a 3-source) extractor, and the intuitive reason for needing the extra independence was that we could not tell which of the output blocks of s_ext was the random one.

[13] A prefix of a random string is random. This flexibility will prove useful in more ways than one.

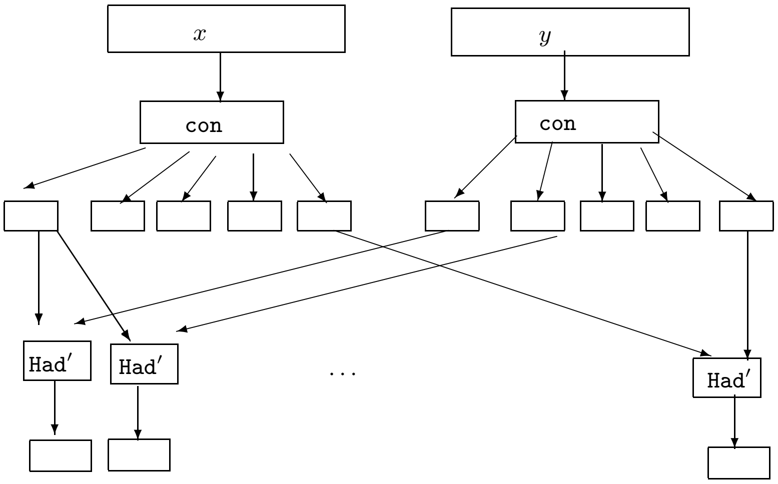

Figure 1: A 2-source somewhere extractor s_ext.

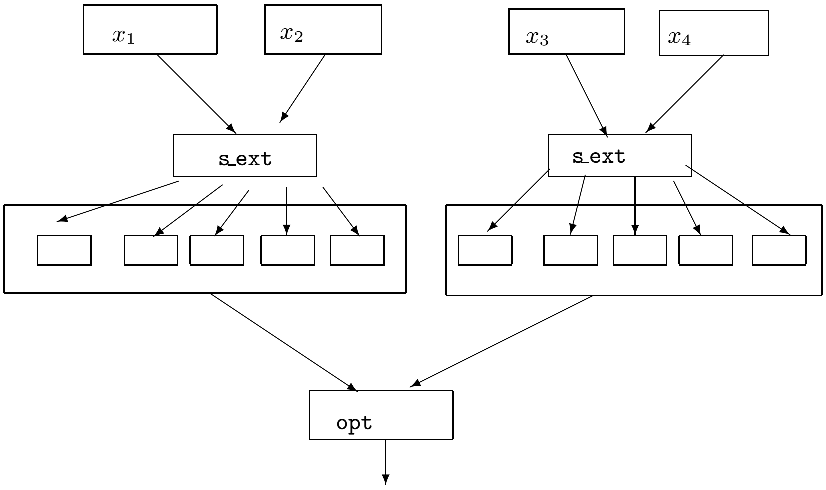

Figure 2: A 4-source extractor 4ext.

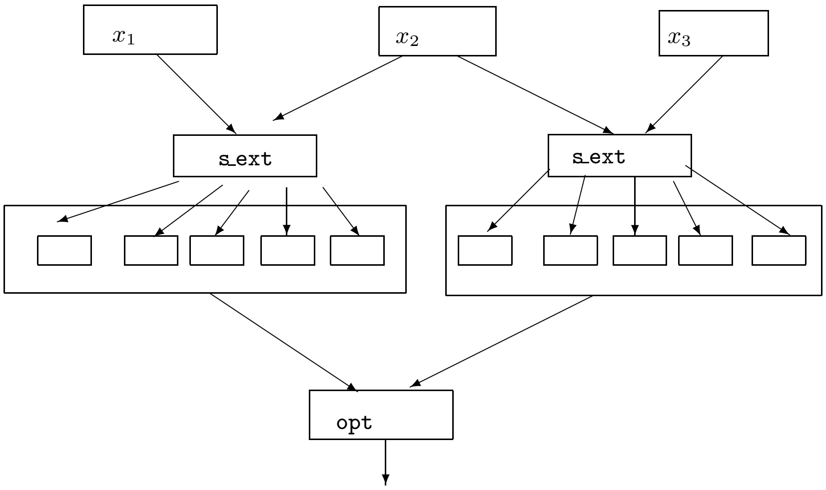

Figure 3: A 3-source extractor 3ext.

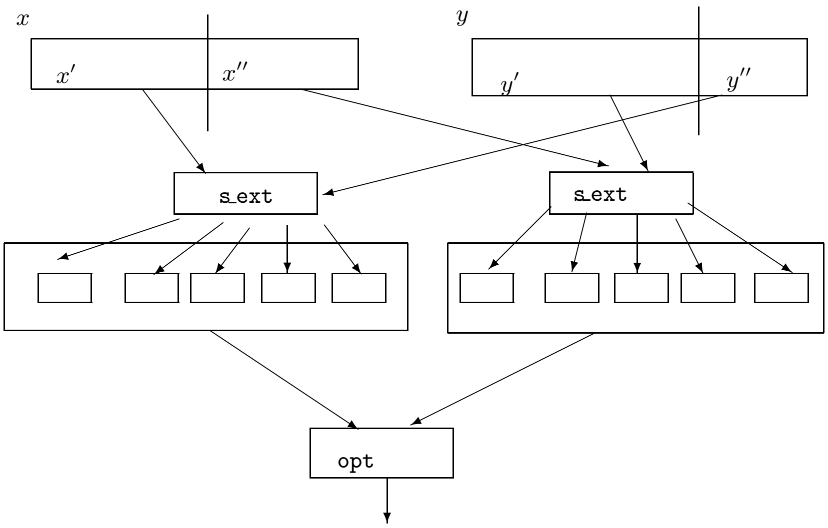

Figure 4: Applying the 4-source extractor to two sources.

To get down to a 2-source (disperser), our general strategy will be to try and "figure out" which of the blocks is random. This requires a "testable" condition on the input, which will point to the right output block. We cannot do that directly with $\texttt{s\_ext}$, so we first develop (using $\texttt{s\_ext}$) a new somewhere extractor $\texttt{s\_ext}'$, for which the random output block is determined by simple entropy conditions on the input sources. The very subtle task of testing these (given only the input sample) is delayed to the next subsection - we now describe $\texttt{s\_ext}'$.

It will be convenient to call now the two independent input δ-sources X and Y. In a nutshell, $\texttt{s\_ext}'$ will consider different partitions of X into $X'X''$ and Y to $Y'Y''$, and for each such partition (X', X'', Y', Y'') apply the 4-source extractor $\texttt{4ext}$ of the previous subsection to obtain $\texttt{4ext}(X', Y'', Y', X'')$ (see Figure 4).

While these 4 sources are not independent, we can use again the same idea used in decreasing 4 sources into 3 of the previous section to try to analyze this construction. As each of the $\texttt{s\_ext}$ components in $\texttt{4ext}$ are *strong*, we can show that fixing X' and Y' to random samples x' and y' (resp.) will suffice to make the output $\texttt{4ext}((x', Y''), (y', X''))$ close to random, as long as we have the following entropy bounds: the four sources $X', Y', X''|X' = x', Y''|Y' = y'$ all have at least ϵn bits of entropy for some constant $\epsilon > 0$.

Another observation is that a one of a constant number of partitions of X and Y *will* satisfy these entropy conditions. This can be shown using the following reasoning. Specifically, let $t = 4/\delta$ and for $i \in [t]$, denote by X_i the i^{th} block of size n/t in X. That is, $X = X_1, \ldots, X_t$. We let $X_{\leq i}$ denote the concatenation of the first i blocks of X (i.e., $X_{\leq i} = X_1, \ldots, X_i$) and let $X_{>i}$ denote the rest of X (i.e., $X_{>i} = X_{i+1}, \ldots, X_n$). We say that i is a "good" index if $X_{\leq i}$ has min-entropy at least $\delta n/4$ and define i_0 be the *smallest* good i. Since the entropy of $X_{\leq i}$ is at most the min-entropy of $X_{\leq i-1}$ plus the length of X_i we get that it is bounded from above by $2 \cdot (\delta n/4) = \delta n/2$. Combining this with the fact that X has min-entropy at least δn, we get that the min-entropy of $X_{>i_0}$ conditioned on fixing $X_{\leq i_0}$ is at least $\delta n/2$. Therefore if we define $X' = X_{\leq i_0}$ and $X'' = X_{>i_0}$ we get the desired partition. In the same way one can partition Y into Y' and Y'' where $Y' = Y_{\leq j_0}$ and $Y'' = Y_{>j_0}$ for some $j_0 \in [t]$. We caution the reader that the effects of conditioning on min-entropy are actually more subtle than what is reflected by this proof sketch. The actual analysis, which is in the full version of this work is thus much more involved than the description here.

Our new somewhere extractor $\texttt{s\_ext}'$ will apply $\texttt{4ext}$ to all t^2 possible ways of defining these four sources from the two given ones X and Y. Specifically, we define $\texttt{s\_ext}'(x, y)_{i,j} = \texttt{4ext}((x_{\leq i}, y_{>j}), (x_{\leq j}, y_{>i}))$. We know that the output block $\texttt{s\_ext}'(x, y)_{i_0,j_0}$ corresponding to i_0 for X and j_0 for Y as defined above *will* be uniformly distributed! This "goodness" will be our "testable" entropy condition.

2.5 Finding the entropy: the "Challenge-Response" mechanism

Ideally, we'd like a test that will find the good partitions and point to the correct output block. Namely, we'd like a function $\texttt{test} : (\{0,1\}^n)^2 \to [t]^2$ such that the output block $\texttt{s\_ext}'(X, Y)_{\texttt{test}(X,Y)}$ is uniform. Note that if we could do that, we'd have a 2-source extractor - more than we promised. Our test will only succeed to point out to the random output block with positive probability (thus yielding our strong notion of a disperser).

Before describing the test, we note the obvious subtlety: $\texttt{test}$ is applied to the *same* sample (x, y) from $X \times Y$ that $\texttt{s\_ext}'$ is applied to! This single sample is supposed to detect the presence of entropy in some parts of the input sources it is coming from, by "looking in the mirror". The fact that this idea can make sense is an important contribution of our paper.

To explain the nature of the test $\texttt{test}$, we oversimplify our problem, and assume we have the following extra gadget to help us.

Temporary unrealistic assumption: We'll assume a deterministic 1-source extractor (called $\texttt{ext}^*$) for any linear entropy sources. It of course seems very strange to assume an object much stronger than what we're trying to construct (indeed so strong that it doesn't exist). Nonetheless we believe it will be a useful didactic tool. Later, we will examine the properties of $\texttt{ext}^*$ that we actually needed in order to obtain the function $\texttt{test}$ and how these properties can be obtained by more realistic gadgets.

We will test each of the prefixes of input parts (potential choices of the first part of the partition X'), namely $X_{\leq 1}, X_{\leq 2}, \cdots, X_{\leq t}$ of X, for containing entropy. To test in the input part $X_{\leq k}$ has entropy, we apply the (hypothetical) extractor $\texttt{ext}^*$ to compute a *challenge* $C_k = \texttt{ext}^*(X_{\leq k})$ of (sufficiently large) constant length c. We will also apply $\texttt{ext}^*$ on the second source Y to generate a string $R = R_1, R_2, \cdots R_t$, where the length of every *response* R_k is c as well. We note that as Y is a δ source, R is essentially a random string of length tc.

We say that the challenge C_k is met by the *response* R_k simply if $R_k = C_k$. We now define the candidate value of the index i' (this index determines the partition of X to $X'X''$ selected by the test $\texttt{test}$) to be $k+1$ for the largest k for which the challenge was met (and default to $i' = 1$ if no challenge was met). We similarly determine the partition of Y by picking j' similarly, reversing the roles of X and Y above. This will give us the pair (i', j') which is the output of the test $\texttt{test}$.

2.6 Analyzing the challenge-response mechanism

Now why does this work, and in what sense? Let us focus on the choice of i' (the argument for j' is analogous). We want to claim that $\texttt{test}$ above produces i' which is the smallest good index for X. While this may fail, we can claim that it holds in a constant probability event in $X \times Y$. Essentially, this event will be defined by conditioning parts of X and Y to fix the outcome of some challenge and response strings, resulting in the "correct" choices made by the test `test`, *and* all independence and entropy requirements holding after these fixings. A simple observation, which will be crucial in the analysis below, is that if X is a random variable on $\{0,1\}^n$, and $f : \{0,1\}^n \to \{0,1\}^c$ is a function with a constant output length c, then if we let z be a "typical" output of $f(X)$ and let X' be X conditioned on $f(X) = z$, then then entropy of X' is only a constant smaller than the entropy of X.

Now, let i_0 be the actual smallest good index for X. For all $k = 1, 2, \cdots, i_0 - 1$ (in that order) we condition X_k in a way that fixes the responses $C_k, k < i_0$ to their most popular values (by the observation above, this reduces the entropy of X by a constant only). Now we also fix the values of the responses $R_k, k < i_0$ so that they meet the respective challenges above. Then we fix set R_k for $k \geq i_0$ to an arbitrary value. Again, fixing all the responses reduces the entropy of Y by only a constant.

We now repeat the same process with j_0, the smallest good index for Y. Call the resulting sub-sources thus generated $\tilde{X}$ and $\tilde{Y}$. Now in the space $\tilde{X} \times \tilde{Y}$ everything we want actually happens.

- $\tilde{X}$ and $\tilde{Y}$ are independent.
- $\tilde{X}$ has constant probability in X (and same for Y), and hence the entropies of $\tilde{X}$ and $\tilde{Y}$ are at least $\delta n - O(1)$.
- This means the the entropy of any subblock of X can also change by at most a constant amount in $\tilde{X}$ (with the same holding for Y). Therefore, $\tilde{X}_{\leq i_0}$, $\tilde{X}_{>i_0}$ is a "good" partition of $\tilde{X}$, with $\tilde{X}_{\leq i_0}$, $\tilde{X}_{>i_0} | \tilde{X}_{\leq i_0}$ both having entropy at least $\delta n/4 - O(1)$. Again, the same holds for $\tilde{Y}$.
- By design, in $\tilde{X}$ the test `test` selects $i' = i_0$ with probability very close to one. This happens since we forced all the first $i_0 - 1$ challenges to be met, and all remaining ones will be missed with very high probability $(1 - t/2^c)$, and so by the definition of `test`, it will select $i' = i_0$. Same happens with $j' = j_0$.

To summarize, with constant probability the sample (x, y) from $X \times Y$ actually lands in the event $\tilde{X} \times \tilde{Y}$, in which case we produce a close to uniform output. Otherwise, we have no guarantee. This is precisely the strong disperser promised.

2.7 Removing the unrealistic assumption

We will be very brief here. The main thing to fix of course, is our assumption that we have available (the impossible) one source extractor $\texttt{ext}^*$. It will be replaced by our favorite 2-source somewhere extractor `s_ext`. This raises several extra issues to deal with, and we touch on the solutions, ignoring many important details.

- When using `s_ext` instead of $\texttt{ext}^*$, what do we use for the second independent input source? The answer is simple - when we used $\texttt{ext}^*$ on part of X, we use (all of) Y as a second source, and vice versa. This raises as variety of issues regarding independence, and regarding preservation of entropy, which did not arise above. Indeed, when we use $\texttt{ext}^*$ to compute a *response* (as opposed to a *challenge*) we will use as input only "tiny" (around $\delta^3 n$ size) parts of X and Y so that these parts can be fixed in the analysis without a significant loss in entropy.
- The output of `s_ext` is not uniform, but rather a constant number of blocks one of which is uniform - how does `test` change? The answer again is expected - the challenges and responses comprise the whole output of `s_ext` (of appropriate lengths). However now a response meets a challenge if *one* of the output blocks of the response equals the (the whole) challenge string. Hence, when computing a response we will output a much larger (although still constant size) string than when computing a challenge. This raises several issues regarding the probabilities with which challenges are met, with bearing on entropies lost, which did not arise above.

This is of course by no means a complete description of our 2-source disperser and its analysis. The complete description and analysis can be found in the appendices.

3. PROOF OF THEOREM 1.3 (BASIC CONDENSER)

Proof. We start by assuming that $n = 3p$ for some prime p. We later explain what we do in the case that n is not of this form. For $x = (x_1, x_2, x_3) \in \{0,1\}^{3p}$ we define $\texttt{bcon}(x) = (x_1, x_2, x_3, x_1 \cdot x_2 + x_3)$.

Let X be a δ-source over $\{0,1\}^{3p}$. Let θ be such that the main lemma of [1] guarantees that for every three independent $\delta(1-10\theta)$-sources A_1, A_2, A_3 over $\mathrm{GF}(2^p)$, the distribution $A_1 \cdot A_2 + A_3$ is $2^{-10\theta\delta p}$-close to having rate at least $\delta(1 + 10\theta)$. (It will be sufficient to choose $\theta = \delta/c$ for some absolute constant c.) We will show that there is a random variable $I = I(X)$ such that $\texttt{bcon}(X)_I$ is $2^{-\theta\delta p}$-close to having rate at least $(1+\theta)\delta$.

We assume without loss of generality that X is a flat source, since it is enough to prove the theorem for such sources, and so we can identify X with the subset of elements on which it is supported. For $i \in [4]$ define $H_i \subseteq \mathrm{GF}(2^p)$ to be the set of "heavy" elements of the distribution $\texttt{bcon}(X)_i$. That is, $H_i = \{y \in \mathrm{GF}(2^p) \mid \Pr[\texttt{bcon}(X)_i = y] \geq 2^{-(1+\theta)\delta p}\}$. Note that for every $i \in [4]$, $|H_i| \leq 2^{(1+\theta)\delta p}$. If $\Pr[\exists_i \text{ s.t. } \texttt{bcon}(X)_i \notin H_i] > 1 - 2^{-\theta\delta p}$ then we're done, since we can define for all but an exponentially small fraction of the x's the index $I(x)$ to be the smallest $i \in [4]$ such that $\texttt{bcon}(x)_i \notin H_i$. Clearly, regardless of how I is defined on these few "bad" x's, we get that $\texttt{bcon}(X)_I$ is exponentially close to having rate $\geq (1 + \theta)\delta$.

Therefore, we may assume that $\Pr[\forall_i \texttt{bcon}(X)_i \in H_i] \geq 2^{-\theta\delta p}$. Since X is a flat source this just means that if

$$X' = \{x = x_1x_2x_3 \in X \mid x \in H_1 \times H_2 \times H_3, x_1 \cdot x_2 + x_3 \in H_4\}$$

then

$$|X'| \geq \frac{2^{3\delta p}}{2^{\theta\delta p}} = 2^{(3-\theta)\delta p} \qquad (1)$$

Note that X' is a subset of $X \cap H_1 \times H_2 \times H_3$. We see that $|H_1| \geq 2^{(1-3\theta)\delta p}$, since otherwise we'll have $|X'| \leq |H_1| \cdot |H_2| \cdot |H_3| < 2^{(1-3\theta)\delta p} 2^{(1+\theta)\delta p} 2^{(1+\theta)\delta p} = 2^{(3-\theta)\delta p}$. Using the same reasoning $|H_2|, |H_3| \geq 2^{(1-3\theta)\delta p}$. We define A_1, A_2, A_3 to be three independent random variables over $\mathrm{GF}(2^p)$ where for $i = 1, 2, 3$, A_i is the flat distribution over the set H_i. Note that for every $i = 1, 2, 3$, the random variable A_i is at least a $\delta(1-3\theta)$-source over $\mathrm{GF}(2^p)$ (since $|H_i| \geq 2^{(1-3\theta)\delta p}$). By Equation 1, it holds that

$$\Pr[A_1 \cdot A_2 + A_3 \in H_4] = \frac{|X'|}{|H_1| \cdot |H_2| \cdot |H_3|} \geq \frac{2^{(3-\theta)\delta p}}{|H_1| \cdot |H_2| \cdot |H_3|} \geq \frac{2^{(3-\theta)\delta p}}{(2^{(1+\theta)p})^3} = 2^{-4\theta\delta p} \quad (2)$$

However, for every $(\delta + 10\theta)$-source Y over $\mathrm{GF}(2^p)$ it holds that

$$\Pr[Y \in H_4] \leq |H_4| 2^{-(1+10\theta)\delta p} \leq 2^{(1+\theta)\delta p} 2^{-(1+10\theta)\delta p} = 2^{-9\theta\delta p} \quad (3)$$

Equations 2 and 3 together imply that the statistical distance of $A_1 \cdot A_2 + A_3$ to *every* $\delta(1+10\theta)$-source Y is at least $2^{-4\theta\delta p} - 2^{-9\theta\delta p} > 2^{-4\theta\delta p - 1}$, contradicting the main lemma of [1].

In the case n is not of the form $n = 3p$ for a prime p, we first pad the input with zeros to length n' for the smallest $n' > n$ such that $n' = 3p$. Using easy number theoretic estimates, $n' < n + 3n^{3/4}$. Since the entropy of X is unchanged by this padding, we get that the rate of X as a random variable over $\{0,1\}^{n'}$ is equal to $\delta \frac{n}{n'} > \frac{\delta}{1+3n^{-1/4}}$. By applying the condenser this rate is going to grow by a multiplicative factor of $1 + \theta = 1 + \Omega(\delta)$ which (since $n^{-1/4} \ll \delta$) will dwarf this factor of $\frac{1}{1+3n^{-1/4}}$. □

We note that the results of [1] are somewhat stronger for a field of *prime* order than for the field $\mathrm{GF}(2^p)$ (i.e., a growth factor of $1 + \Omega(1)$ instead of $1 + \Omega(\delta)$). However, we don't use these stronger results since we are not aware of a deterministic polynomial-time uniform algorithm that on input n outputs a prime of length n bits (or at least a prime with length in the range $[n, n + o(n)]$).[14]

4. REFERENCES

[1] B. Barak, R. Impagliazzo, and A. Wigderson. Extracting randomness from few independent sources. In *Proc. 45th FOCS* , 2004.

[2] 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.

[3] M. Ben-Or and N. Linial. Collective coin flipping, robust voting schemes and minima of Banzhaf values. In *Proc. 26th FOCS*, pages 408–416, 1985.

[4] E. Ben-Sasson, S. Hoory, E. Rosenman, and S. Vadhan. Personal communication, 2001.

[5] M. Blum. Independent unbiased coin flips from a correlated biased source: A finite state Markov chain. In *Proc. 25th FOCS*, pages 425–433, 1984.

[6] J. Bourgain. More on the Sum-Product Phenomenon in Prime Fields and its Applications, 2005. Unpublished manuscript.

[14]Such an algorithm does exist assuming the Extended Riemann Hypothesis.

[7] J. Bourgain, N. Katz, and T. Tao. A sum-product estimate in finite fields, and applications. Arxiv technical report, `http://arxiv.org/abs/math.CO/0301343`, 2003. To appear in GAFA.

[8] 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.

[9] B. Chor and O. Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. In *Proc. 26th FOCS*, pages 429–442, 1985.

[10] 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, 1985.

[11] A. Cohen and A. Wigderson. Dispersers, deterministic amplification, and weak random sources. In *Proc. 30th FOCS*, pages 14–19. IEEE, 1989.

[12] Y. Dodis, A. Elbaz, R. Oliveira, and R. Raz. Improved randomness extraction from two independent sources. In *Proc. of 8th RANDOM*, 2004.

[13] Y. Dodis and J. Spencer. On the (non)universality of the one-time pad. In *Proc. 43rd FOCS*, pages 376–388. IEEE, 2002.

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

[15] A. Gabizon, R. Raz, and R. Shaltiel. Deterministic extractors for bit-fixing sources by obtaining an independent seed. In *Proc. 45th FOCS*, 2004.

[16] Kamp and Zuckerman. Deterministic extractors for bit-fixing sources and exposure-resilient cryptography. In *Proc. 44th FOCS*. IEEE, 2003.

[17] 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.

[18] 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.

[19] Mossel and Umans. On the complexity of approximating the VC dimension. *J. Comput. Syst. Sci.*, 65, 2002.

[20] P. Pudlak. On explicit ramsey graphs and estimates on the numbers of sums and products, 2005. Unpublished manuscript.

[21] P. Pudlák and V. Rödl. Pseudorandom sets and explicit constructions of ramsey graphs, 2004. Submitted for publication.

[22] R. Raz. Extractors with weak random seeds, 2004. Appears in these proceedings.

[23] O. Reingold, R. Shaltiel, and A. Wigderson. Extracting randomness via repeated condensing. In *Proc. 41st FOCS*, pages 22–31, 2000.

[24] M. Santha and U. V. Vazirani. Generating quasi-random sequences from slightly-random sources. In *Proc. 25th FOCS*, pages 434–440, 1984.

[25] R. Shaltiel. Recent developments in extractors. *Bulletin of the European Association for Theoretical Computer Science*, 2002. Available from `http://www.wisodm.weizmann.ac.il/~ronens`.

[26] A. Ta-Shma, C. Umans, and Z. Zuckerman. Loss-less condensers, unbalanced expanders, and extractors. In *Proc. 42nd FOCS*, pages 143–152. IEEE, 2001.

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

[28] U. Vazirani. Strong communication complexity or generating quasi-random sequences from two communicating semi-random sources. *Combinatorica*, 7, 1987. Preliminary version in STOC' 85.

[29] J. von Neumann. Various techniques used in connection with random digits. *Applied Math Series*, 12:36–38, 1951.

[30] D. Zuckerman. General weak random sources. In *Proc. 31st FOCS*, pages 534–543. IEEE, 1990.