2015 IEEE 56th Annual Symposium on Foundations of Computer Science

Local Correlation Breakers and Applications to Three-Source Extractors and Mergers

Gil Cohen
Department of Computer Science and Applied Mathematics
Weizmann Institute of Science
Rehovot, Israel
gil.cohen@weizmann.ac.il

Abstract

We introduce and construct a pseudorandom object which we call a local correlation breaker (LCB). Informally speaking, an LCB is a function that gets as input a sequence of r (arbitrarily correlated) random variables and an independent weak-source. The output of the LCB is a sequence of r random variables with the following property. If the i'th input random variable is uniform then the i'th output variable is uniform even given a bounded number of any other output variables. That is, an LCB uses the weak-source to break local correlations between random variables. Our construction of LCBs has applications to three-source extractors, mergers with weak-seeds, and a variant of non-malleable extractors, that we introduce.

Keywords

multi-source extractors; mergers; non-malleable extractors;

I. Introduction

A central theme in pseudorandomness concerns the design of efficient algorithms that transform one or more sources of randomness to a source with a desired property. From the applications end, the most typical desired property from a source of randomness is simply for it to be uniformly distributed over some ambient support. Informally speaking, extractors accomplish exactly this task as they produce truly random bits given a sample from a defective source of randomness.

Constructing extractors of various types is challenging and many constructions in the literature rely on auxiliary pseudorandom primitives such as mergers and condensers. These auxiliary objects either have weaker guarantee on the output or further assumptions on the input, so they are not fit to serve as off-the-shelf objects and are therefore more intrinsic to the subfield of pseudorandomness. Nevertheless, such objects are very useful as building blocks for the construction of other pseudorandom objects.

When constructing pseudorandom objects such as extractors, one is often faced with the problem of correlations between random variables. Namely, at some point in the construction a sequence of random variables $X_1, \ldots, X_r$ is obtained, such that one or more of these random variables is "well-behaved", yet the correlations between the variables prevent one from proceeding with the construction and analysis.

In this work we identify and face the problem above. We introduce and construct a pseudorandom primitive that we call a *local correlation breaker* (LCB for short) that, as its name suggests, allows one to "break" local correlations between random variables. We further present applications of LCBs to the construction of three-source extractors, mergers with weak-seeds, and a variant of non-malleable extractors. As LCBs allows one to face the typical scenario above, we believe LCBs will find further applications in the future.

The "well-behaved" property that we consider in this work is being uniformly distributed. Ideally, for some locality parameter t, an LCB would be an algorithm that gets as input a sequence of r (arbitrarily correlated) random variables, and outputs a sequence of r random variables with the following property: If the i^{th} input random variable is uniform then the i^{th} output variable is uniform even given any other $t-1$ output variables.

Unfortunately, regardless of efficiency, no deterministic algorithm can accomplish the task above. A natural suggestion would be to consider seeded-LCBs, namely, LCBs that have a short auxiliary string of truly random

0272-5428/15 $31.00 © 2015 IEEE
DOI 10.1109/FOCS.2015.57

bits (that is independent of the input variables). Although this suggestion is natural and appealing, given our applications in mind, we consider a different (and more challenging) variant where the auxiliary source of randomness is a weak-source of randomness. For the formal definition of LCBs we make use of standard definitions from the literature such as min-entropy, statistical distance, and (n,k)-weak-sources (see the Preliminaries).

Definition 1.1 (Local correlation breakers): A *t-local correlation breaker (t-LCB)* for min-entropy k, with error ε, is a function

$$\mathsf{LCB}\colon \left(\{0,1\}^{\ell}\right)^{r} \times \{0,1\}^{n} \to \left(\{0,1\}^{m}\right)^{r},$$

with the following property. Let $X = (X_1, \ldots, X_r)$ be a sequence of random variables, each supported on $\{0,1\}^{\ell}$. Let Y be an independent (n,k)-weak-source. Denote the output $\mathsf{LCB}(X,Y)$ by $Z_1, \ldots, Z_r$, where each Z_i is supported on $\{0,1\}^{m}$. Let $g \in [r]$ be such that X_g is uniform. Let $I \subseteq [r] \setminus \{g\}$ be any set of size $t-1$. Then,

$$(Z_g, \{Z_i\}_{i\in I}) \approx_{\varepsilon} (U_m, \{Z_i\}_{i\in I}).$$

Our first result is an explicit construction of LCBs. For simplicity, the theorem below is stated for constant error.

Theorem 1.2 (Explicit LCBs; informal statement): For all integers n, r, t, there exists an explicit t-local correlation breaker $\mathsf{LCB}\colon \left(\{0,1\}^{\ell}\right)^{r} \times \{0,1\}^{n} \to \left(\{0,1\}^{m}\right)^{r}$ with

$$\ell = O\left(t^2 \cdot \log(nr) \cdot \log r\right),$$
$$m = \Omega\left(\ell/(t \cdot \log r)\right),$$

for entropy $k = O\left(t \cdot \log(r) \cdot \log\left(r \log n\right)\right)$.

We emphasize the surprising fact that, in our construction, the dependence of the entropy k in n is double-logarithmic. One can show that a random function has the same dependence of k in n, and so it is plausible that this is the right dependence. For a complete and formal statement of Theorem 1.2, see the full version of this extended abstract.

A pseudorandom object related to LCBs appears (implicitly) in the analysis of Li's multi-source extractor [Li13a]. The difference between LCBs and Li's pseudorandom object is that the latter only guarantees that an output variable Z_g that corresponds to a uniform input variable X_g is statistically-close to uniform given output variables that correspond to $t-1$ other *uniform* input variables. In other words, in Li's pseudorandom object, the set I in Definition 1.1 is assumed to contain indices only of uniform input variables. For the applications we consider, it is crucial that Z_g is close to uniform even given output variables $\{Z_i\}$ that correspond to possibly non-uniform input variables.

Our proof builds on the work of [Li13a], together with some new ideas required so to guarantee the stronger property. In the full version of this paper we give a high-level comparison between the ideas used in [Li13a] and those used in our construction of LCBs. In terms of parameters, Theorem 1.2 gives LCBs with somewhat better parameters compared to Li's pseudorandom object (even though the guarantee is stronger).

We believe that LCBs are natural pseudorandom primitives as they allow one to face the recurrent difficulty of having correlation between random variables using (very) weak-sources of randomness. Next we exemplify the usefulness of LCBs by presenting several applications. Further, in a subsequent work, Chattopadhyay, Goyal, and Li [CGL15] applied ideas from our construction of LCBs for the construction of non-malleable codes and extractors (see Section I-C). The latter was then used as a key component in the recent breakthrough construction of two-source extractors by Chattopadhyay and Zuckerman [CZ15].

A. Three-source extractors with a double-logarithmic entropy source

Chor and Goldreich [CG88] introduced the notion of two-source extractors. Informally speaking, a function $f\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$ is called a *two-source extractor* if for any two independent sources X, Y over $\{0,1\}^n$, with sufficient min-entropy, it holds that $f(X,Y)$ is statistically-close to uniform. By a standard

probabilistic argument, one can prove the existence of a two-source extractor for any entropies k_1, k_2 such that $\min(k_1, k_2) > \log n + O(1)$.

Chor and Goldreich [CG88] gave an explicit construction of a two-source extractor for any entropies k_1, k_2 such that $k_1 + k_2 > (1 + \delta) \cdot n$, where $\delta > 0$ is an arbitrarily small constant. In particular, one can take $k_1 = k_2 > (1/2+\delta) \cdot n$, for any constant $\delta > 0$. This construction is far from optimal (ignoring the computational aspect). Nevertheless, it took almost 20 years before any improvement was made. Raz [Raz05] gave an explicit construction of a two-source extractor for sources with entropies k_1, k_2, with $k_1 = O(\log n)$ and $k_2 > (1/2+\delta) \cdot n$, where $\delta > 0$ is an arbitrarily small constant. An incomparable result was obtained by Bourgain [Bou05] who constructed a two-source extractor for entropies $k_1 = k_2 > (1/2 - \alpha) \cdot n$, where $\alpha > 0$ is some (small) universal constant.

Given the difficulty of explicitly constructing two-source extractors for low entropy, a significant research effort was directed towards the construction of t-source extractors for $t > 2$. The next natural goal is constructing three-source extractors. A simple probabilistic argument can be used to prove the existence of an extractor for three independent $(n, \log(n)/2 + O(1))$-weak-sources. Barak *et al.* [BKS$^+$05] gave an explicit construction of a three-source extractor, where the entropy of each of the sources is δn, for any constant $\delta > 0$. This was improved by Raz [Raz05], who requires only one of the sources to have entropy δn, while the other two sources can have entropy $O(\log n)$. Here, again, $\delta > 0$ is an arbitrarily small constant. Raz's extractor supports a constant error, and in a subsequent work, Rao [Rao09] showed how to support exponentially small error, assuming the second and third sources have entropy $O(\log^4 n)$. Furthermore, Rao [Rao09] constructed a three-source extractor, where the entropy of each of the sources is $n^{0.9}$. This was later improved by Li [Li11] to $n^{1/2+\delta}$, where $\delta > 0$ is an arbitrarily small constant.

In a recent breakthrough, Li [Li15] constructed a three-source extractor for poly-logarithmic entropy. This exciting result sets the next natural goal in multi-source extractors on improving the constructions of two-source extractors by Raz [Raz05] and Bourgain [Bou05]. Towards this goal, as an application of LCBs, we construct a three-source extractor where one of the sources is only assumed to have double-logarithmic entropy.

Theorem 1.3 (Explicit three-source extractors; informal statement): For any integer n and $\delta > 0$, there exists an explicit three-source extractor $\mathsf{3Ext}\colon (\{0,1\}^n)^3 \to \{0,1\}^m$ for entropies

$$\begin{aligned} k_1 &= \delta n, \\ k_2 &= \mathrm{poly}(1/\delta) \cdot \log n, \\ k_3 &= \mathrm{poly}(1/\delta) \cdot \log\log n, \end{aligned}$$

with $m = \mathrm{poly}(1/\delta) \cdot \log n$ output bits.

The extractor in Theorem 1.3 is incomparable with the extractor of [Li15] and improves Raz's and Rao's three-source extractors [Raz05], [Rao09] which assume the third source has entropy $\Omega(\log n)$. As the third source fed to our three-source extractor is required to have a tantalizingly low entropy, we hope that further ideas can be used to eliminate the need for this third source altogether. A formal statement of Theorem 1.3 and its proof appear in the full version of this work.

Improved three-source extractors for poly-logarithmic entropy: As mentioned, Li [Li15] constructed a three-source extractor for poly-logarithmic entropy. More precisely, the entropy required by Li's construction is $O(\log^{12} n)$. For his construction, Li uses a pseudorandom object that is related to LCBs, introduced in [Li13a], as well as the merger with weak-seeds of [BRSW12]. As our construction of LCBs has better parameters than Li's related pseudorandom object and since our merger with weak-seeds improves that of [BRSW12] (see the following section), by using our results as building blocks in Li's three-source extractor, one can obtain a three-source extractor for a somewhat lower entropy of $\widetilde{O}(\log^c n)$, where $c < 12$ is some constant. By a short calculation, one can show that $c = 7$. Nevertheless, this calculation was not verified as carefully as the proofs in this paper, and should be trusted accordingly.

B. Mergers with weak-seeds

Motivated by the construction of seeded extractors, Ta-Shma [TS96] introduced the notion of a merger. Informally speaking, a merger is a function that gets as input a sequence of (arbitrarily correlated) random variables, at least one of which is uniform. The goal of a merger is to "merge" the random variables into a single random variable that is statistically-close to uniform. [1] It is not hard to show that randomness is a necessity for merging.

Constructing mergers with short seeds (namely, short strings that are uniform and independent of the random variables we wish to merge) has been studied in several works [TS96], [LRVW03], [Raz05], [DS07], [Zuc07], [DR08], [DW11], [DKSS09]. The state of the art construction of Dvir and Wigderson [DW11] merges r random variables, supported on $\{0,1\}^\ell$, using a seed of length $O(\log(r\ell))$. An incomparable result was obtained by Dvir, Kopparty, Saraf and Sudan [DKSS09], who use a seed of length $O(\log(r)/\delta)$ to output a string that has entropy-rate $1-\delta$. As a building block for their two-source disperser, Barak, Rao, Shaltiel and Wigderson [BRSW12] constructed, what we call, mergers with weak-seeds. [2]

Definition 1.4 (Mergers with weak-seeds): A *merger with weak-seeds* for entropy k, with error ε, is a function

$$\mathsf{Merg}\colon \left(\{0,1\}^\ell\right)^r \times \{0,1\}^n \to \{0,1\}^m,$$

with the following property. Let $X = (X_1, \ldots, X_r)$ be a sequence of random variables, supported on $\{0,1\}^\ell$, such that at least one of them is uniform. Let Y be an independent (n,k)-weak-source. Then, $\mathsf{Merg}(X,Y) \approx_\varepsilon U_m$.

In [BRSW12], a construction of a merger with weak-seeds is given, assuming $k = \ell > \Omega(r^2) + \mathrm{polylog}(n)$. [3] A probabilistic argument can be used to show that there exists a merger with weak-seeds for parameters

$$\begin{aligned}\ell &= \log n + O(1),\\ k &= \log r + \log\log n + O(1).\end{aligned}$$

In particular, the entropy k is only required to be double-logarithmic in n.

We note that constructing mergers with weak-seeds given an LCB is trivial. Indeed, one can apply an r-LCB to $X_1, \ldots, X_r$ and Y so to obtain random variables $Z_1, \ldots, Z_r$. The output of the merger is simply the XOR of all Z_i's. To see that this reduction works, note that if X_g is uniform then, by the guarantee of the LCB, Z_g is statistically-close to uniform even given all other Z_i's. Therefore, the XOR of all Z_i's is statistically-close to uniform. We use Theorem 1.2 with this simple idea (together with a bit more work so to improve the output length) and obtain the following result.

Theorem 1.5 (Explicit mergers with weak-seeds; informal statement): For all integers n, r, there exists an explicit merger with weak-seeds $\mathsf{Merg}\colon \left(\{0,1\}^\ell\right)^r \times \{0,1\}^n \to \{0,1\}^m$, with

$$\begin{aligned}\ell &= O\left(r^2 \cdot \log(r) \cdot \log(nr)\right),\\ k &= O\left(r \cdot \log(r) \cdot \log(r \cdot \log n)\right),\\ m &= \Omega(\ell/r).\end{aligned}$$

A formal statement of Theorem 1.5 and its proof appear in the full version of this paper. In Section V we give a warm up for the proof of Theorem 1.5, and show how to merge $r = 3$ rows of length $\ell = O(\log n)$ using an independent weak-source with entropy $k = O(\log\log n)$ (see Theorem 5.1). This "toy-example" also demonstrates most of the ideas used in the proof of Theorem 1.2.

The merger of Barak *et al.* [BRSW12] and ours are incomparable. On one hand, the merger of [BRSW12] works even if one of the rows has min-entropy rate $1 - o(1)$. On the other hand, Theorem 1.5 has a quadratically

[1] Variants of mergers (which are also called mergers in the literature) assume that one of the random variables is not necessarily uniform, yet has high entropy-rate.

[2] In [BRSW12] this object is called an extractor for a general source and a somewhere-random source.

[3] In fact, the construction of [BRSW12] works even assuming one of the X_i's has entropy-rate $1 - o(1)$.

improved dependence of k in r, and more importantly, an exponentially improved dependence of k in n, which matches the probabilistic construction. This feature allows us to obtain a three-source extractor with double-logarithmic entropy source. Moreover, we believe our construction and analysis are somewhat simpler and more intuitive than the construction of [BRSW12] which uses a completely different set of ideas.

C. Two-source non-malleable extractors

Non-malleable extractors were introduced by Dodis and Wichs [DW09], motivated by the classical problem of privacy amplification. Informally speaking, a non-malleable extractor is a seeded extractor with a very strong pseudorandom property – the output of a non-malleable extractor obtained using a typical seed does not reveal information about the output that one would get using any different seed. More formally, a function $\mathsf{NMExt}\colon \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is called a *non-malleable extractor* for entropy k, if for any function $A\colon \{0,1\}^d \to \{0,1\}^d$ with no fixed points (that is, $A(s) \neq s$ for all $s \in \{0,1\}^d$), and for any (n,k)-weak-source X, it holds that $\mathsf{NMExt}(X,S)$ is statistically-close to uniform even given $(S, \mathsf{NMExt}(X, A(S)))$. Here S is uniformly distributed over $\{0,1\}^d$ and independent of X. In [CRS14], the notion of a t-non-malleable extractor was introduced. This is a seeded extractor for which $\mathsf{NMExt}(X,S)$ is statistically-close to uniform even conditioned on $(S, \{\mathsf{NMExt}(X, A_i(S))\}_{i=1}^t)$, where all A_i's have no fixed points.

Dodis and Wichs [DW09] showed that a random function of the form $\mathsf{NMExt}\colon \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is, with high probability, a non-malleable extractor for entropy $k = 2m + \log d$, with seed length $d = \log n + O(1)$. Constructing non-malleable extractors was considered to be a challenging problem, and several explicit constructions of non-malleable extractors [DLWZ14], [CRS14], [Li12a], [Li12b] were only able to support entropy roughly $n/2$. Following the publication of this work, and building on ideas used in the construction of LCBs, Chattopadhyay, Goyal, and Li [CGL15] significantly improved these results, and constructed non-malleable extractors for entropy $O(\log^2 n)$ with seed length $O(\log^2 n)$.

Although the non-malleable extractor by Chattopadhyay *et al.* [CGL15] supports polylogarithmic entropy, the fact that the seed length is quadratic in $\log n$ does not allow for an efficient enumeration of the extractor applied to all seeds with a given sample. Such enumeration is a basic step in many constructions of extractors [Rao09], [BRSW12], [Li11], [Li13b], [Li13a], [Li15], including this work.

Towards the goal of constructing non-malleable extractors for polylogarithmic entropy with logarithmic seed length, we consider a relaxation of non-malleable extractors, which we call *two-source non-malleable extractors*. A two-source non-malleable extractor is a function $f\colon (\{0,1\}^n)^2 \times \{0,1\}^d \to \{0,1\}^m$, with the following property. For any two independent (n,k)-weak-sources X, Y, it holds that $f(X,Y,S)$ is statistically-close to uniform even given $(S, f(X,Y,A(S)))$, where the seed S is uniformly distributed over $\{0,1\}^d$ and is independent of (X,Y). More generally, we consider two-source t-non-malleable extractors, which are defined similarly to t-non-malleable extractors.

We recall that computational aspects aside, if one has access to two independent weak-sources, then one can output a string that is statistically-close to uniform by applying a two-source extractor, without any auxiliary seed. However, currently we do not know how to efficiently construct two-source extractors for poly-logarithmic entropy or even for entropy-rate 0.49, and two-source non-malleable extractors can be viewed as a relaxation both of non-malleable extractors and of two-source extractors. In particular, one can view a two-source non-malleable extractor with seed length d as a collection of 2^d two-source extractors with the following property: for any two independent weak-sources X, Y with sufficiently high entropy, most extractors in the collection are two-source extractors for X, Y, and moreover, the output of a "good" two-source extractor in the collection applied to X, Y is independent of the output of any other $t-1$ extractors from the collection applied to the same samples.

Building on our construction of LCBs from Theorem 1.2 and on ideas from [Li15], we construct a two-source non-malleable extractor for poly-logarithmic entropy, with a seed of logarithmic length.

Theorem 1.6 (Explicit two-source t-non-malleable extractors; informal statement): For all integers n, t, there exists an explicit two-source t-non-malleable extractor $\mathsf{2NMExt}\colon (\{0,1\}^n)^2 \times \{0,1\}^d \to \{0,1\}^m$ for entropy $O(t^2 \cdot \log^2 n)$, with $d = O(\log n)$ and $m = \Theta(t \cdot \log n)$ output bits.

It is worth noting that not only the seed length d in Theorem 1.6 is logarithmic in n but it is also independent of t.

D. Organization of this paper

The paper is organized as follows. In Section II we give formal definitions and state some of the results from previous works that we use. In Section III we prove some technical lemmata on probabilistic processes that appear frequently throughout the paper. In Section IV we present a restricted version of look-ahead extractors. Due to lack of space, in this extended abstract we do not give full proofs for our theorems. Nevertheless, in Section V we construct a merger with weak-seeds for three random variables. This demonstrates most of the ideas used in the constructions of our mergers with weak-seeds and LCBs.

II. Preliminaries

The logarithm in this paper is always taken base 2. For every natural number $n \geq 1$, define $[n] = \{1, 2, \ldots, n\}$. For a string $x \in \{0,1\}^n$ and an integer $1 \leq s \leq n$, we write $x|_s$ for the length s prefix of x. For an $r \times \ell$ matrix x and $1 \leq s \leq \ell$, we let $x|_s$ denote the matrix composed of the s leftmost columns of x. For $i \in [r]$, we denote by x_i the i^{th} row of x. Throughout the paper we almost always avoid the use of floor and ceiling in order not to make the equations cumbersome.

Random variables and distributions: We sometimes abuse notation and syntactically treat random variables and their distribution as equal, specifically, we denote by U_m a random variable that is uniformly distributed over $\{0,1\}^m$. Furthermore, if U_m appears in a joint distribution (U_m, X) then U_m is independent of X. When m is clear from context, we omit it from the subscript and write U.

Let X, Y be two random variables. We say that Y is a *deterministic function of* X if the value of X determines the value of Y. Namely, there exists a function f such that $Y = f(X)$. Let $X, Y, Z_1, \ldots, Z_r$ be random variables. We introduce the following shorthand notation and write $(X, Z_1, \ldots, Z_r) \approx_\varepsilon (Y, \cdot)$ for $(X, Z_1, \ldots, Z_r) \approx_\varepsilon (Y, Z_1, \ldots, Z_r)$.

Statistical distance: The *statistical distance* between two distributions X, Y on the same domain D is defined by

$$\mathsf{SD}(X, Y) = \max_{A \subseteq D} \{ |\Pr[X \in A] - \Pr[Y \in A]| \}.$$

If $\mathsf{SD}(X, Y) \leq \varepsilon$ we write $X \approx_\varepsilon Y$ and say that X is ε-close to Y.

Min-entropy: The *min-entropy* of a random variable X is defined by

$$H_\infty(X) = \min_{x \in \sup(X)} \log_2 \left(\frac{1}{\Pr[X = x]} \right).$$

If X is supported on $\{0,1\}^n$, we define the *min-entropy rate* of X by $H_\infty(X)/n$. In such case, if X has min-entropy k or more, we say that X is an (n,k)-weak-source.

Somewhere-random sources: A random variable X that has the form of an $r \times \ell$ matrix is called a *somewhere-random source* if there exists $g \in [r]$ such that X_g is uniformly distributed over $\{0,1\}^\ell$. In this case we say that X_g is a *good* row of X. We think of a somewhere-random source as a sequence of (arbitrarily correlated) random variables given by its rows $X_1, \ldots, X_r$.

Extractors and condensers

Definition 2.1 (Seeded extractors): A function $\mathsf{Ext}\colon \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is called a *seeded extractor* for entropy k, with error ε, if for any (n,k)-weak-source X it holds that $\mathsf{Ext}(X, S) \approx_\varepsilon U_m$, where S is uniformly distributed over $\{0,1\}^d$ and is independent of X. We say that Ext is a *strong seeded-extractor* if $(\mathsf{Ext}(X, S), S) \approx_\varepsilon (U_m, U_d)$, where X and S are as above.

Throughout the paper we make an extensive (black-box) use of the following strong seeded-extractor by Guruswami, Umans and Vadhan [GUV09].

Theorem 2.2 ([GUV09]): For all positive integers n, k and $\varepsilon > 0$, there exists an efficiently-computable strong seeded-extractor $\mathsf{Ext}\colon \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ for entropy k, with error ε, seed length $d = \log n + O(\log(k/\varepsilon))$, and $m = 0.99 \cdot k$ output bits.

Basic lemmata in probability

Throughout the paper we make a frequent use of the following simple and well-known lemmata.

Lemma 2.3: Let X, Y be two independent random variables on a common domain D. Let f be a function with domain D. Then, $\mathsf{SD}\left(f(X), f(Y)\right) \leq \mathsf{SD}\left(X, Y\right)$. Moreover, the inequality above holds also for f which is a random function, where the internal randomness of f is independent of (X, Y).

Lemma 2.4: For all random variables X, Y, Z, it holds that

$$\mathsf{SD}\left((X,Y),(Z,Y)\right) = \mathop{\mathbf{E}}_{y\sim Y}\left[\mathsf{SD}\left((X \mid Y=y),(Z \mid Y=y)\right)\right].$$

Lemma 2.5: Let X, Y, Z be random variables such that X is independent of Y and Z is independent of Y. Then, $\mathsf{SD}\left((X,Y),(Z,Y)\right) = \mathsf{SD}\left(X,Z\right)$. In particular, if X is supported on $\{0,1\}^a$ then $\mathsf{SD}\left((X,Y),(U_a,Y)\right) = \mathsf{SD}\left(X,U_a\right)$.

Lemma 2.6: Let X, Y, Z be random variables such that for any $y \in \sup(Y)$, the random variables $(X \mid Y = y)$ and $(Z \mid Y = y)$ are independent. Assume that X is supported on $\{0,1\}^a$. Then,

$$\mathsf{SD}\left((X,Y,Z),(U_a,Y,Z)\right) = \mathsf{SD}\left((X,Y),(U_a,Y)\right).$$

Lemma 2.7: Let X, Z be random variables on a common domain. Let Y be some random variable. Then, $\mathsf{SD}\left(X,Z\right) \leq \mathsf{SD}\left((X,Y),(Z,Y)\right)$.

Lemma 2.8: Let X, Y be two random variables on a common domain D. Let $f\colon D \to R$ be a function with non-negative range, that is, $f(z) \geq 0$ for all $z \in D$. Then,

$$\left|\mathop{\mathbf{E}}_{x\sim X}\left[f(x)\right] - \mathop{\mathbf{E}}_{y\sim Y}\left[f(y)\right]\right| \leq \max_{z\in D}\left|f(z)\right| \cdot \mathsf{SD}(X,Y).$$

Lemma 2.9 ([Li12b], Lemma 3.20): Let (X, Y) be a joint distribution. Let Z be a random variable with the same range as X. Then, there exists a joint distribution (Z, Y) such that $\mathsf{SD}\left((X,Y),(Z,Y)\right) = \mathsf{SD}(X,Z)$.

Average conditional min-entropy

Definition 2.10: Let X, W be two random variables. The *average conditional min-entropy* of X given W is defined as

$$\begin{aligned}\widetilde{H}_\infty(X \mid W) &= -\log_2\left(\mathop{\mathbf{E}}_{w\sim W}\left[\max_x \Pr\left[X = x \mid W = w\right]\right]\right)\\ &= -\log_2\left(\mathop{\mathbf{E}}_{w\sim W}\left[2^{-H_\infty(X\mid W=w)}\right]\right).\end{aligned}$$

Lemma 2.11 ([DORS08]): Let X, Y, Z be random variables such that Y has support size at most 2^ℓ. Then,

$$\widetilde{H}_\infty(X \mid (Y,Z)) \geq \widetilde{H}_\infty((X,Y) \mid Z) - \ell \geq \widetilde{H}_\infty(X \mid Z) - \ell.$$

In particular, $\widetilde{H}_\infty(X \mid Y) \geq H_\infty(X) - \ell$.

Lemma 2.12 ([DORS08]): For any two random variables X, Y and any $\varepsilon > 0$, it holds that

$$\Pr_{y\sim Y}\left[H_\infty(X \mid Y=y) < \widetilde{H}_\infty(X \mid Y) - \log(1/\varepsilon)\right] \leq \varepsilon.$$

We also need the following simple lemma.

Lemma 2.13: Let X, Y, Z be random variables such that for any $y \in \sup(Y)$ it holds that $(X \mid Y = y)$ and $(Z \mid Y = y)$ are independent. Then, $\widetilde{H}_\infty(X \mid (Y,Z)) = \widetilde{H}_\infty(X \mid Y)$. In particular, if X and Z are independent then $\widetilde{H}_\infty(X \mid Z) = H_\infty(X)$.

III. (L, R)-HISTORIES

In this section we introduce the notion of an (L, R)-*history* and some technical lemmata concerning it that we use repeatedly throughout the paper.

Definition 3.1 ((L, R)-histories): Let L, R be two independent random variables. A sequence of random variables $\mathcal{H} = (H_t, H_{t-1}, \ldots, H_1)$ is called an (L, R)-*history* if for any $i \in [t]$, H_i is either a deterministic function of $H_{i-1}, \ldots, H_1, L$ or otherwise H_i is a deterministic function of $H_{i-1}, \ldots, H_1, R$.

Some remarks and notations:

- Throughout the paper we assume that each H_i is supported on bit strings of some common length, which we can then denote by $|H_i|$.
- We note that if H_{i+1}, H_i are two consecutive random variables in some (L, R)-history, such that H_i is a deterministic function of $H_{i-1}, \ldots, H_1, L$ (resp. $H_{i-1}, \ldots, H_1, R$) and H_{i+1} is a deterministic function of $H_i, \ldots, H_1, L$ (resp. $H_{i-1}, \ldots, H_1, R$), then one can replace H_{i+1}, H_i by a single random variable which is their joint distribution. This yields a new (L, R)-history. We allow ourselves to apply this operation freely during the proofs.
- Given two (L, R)-histories $\mathcal{H} = (H_t, \ldots, H_1)$ and $\mathcal{H}' = (H'_{t'}, \ldots, H'_1)$, one can consider the (L, R)-history which is the concatenation of $\mathcal{H}, \mathcal{H}'$, namely, $H'_{t'}, \ldots, H'_1, H_t, \ldots, H_1$. When we do not want to refer to the random variables in $\mathcal{H}$ but do want to refer to the random variables in $\mathcal{H}'$ (which is quite frequent), we write this concatenated (L, R)-history as $(H'_{t'}, \ldots, H'_1, \mathcal{H})$.

The following lemma states that conditioned on any fixing of an (L, R)-history, the random variables L, R remain independent. We omit the proof, which is done by a straightforward induction.

Lemma 3.2: Let L, R be two independent random variables, and let $\mathcal{H}$ be an (L, R)-history. Then, for any $h \in \mathrm{sup}(\mathcal{H})$, the random variables $(L \mid \mathcal{H} = h)$ and $(R \mid \mathcal{H} = h)$ are independent.

In the rest of this section we state and prove two technical lemmata for (L, R)-histories. Before giving the formal statement of the first lemma, we present the lemma in an informal manner so to give some intuition about what the lemma aims to abstract. A common scenario in our proofs is the following. Let L, R be two independent random variables. We think of L, R as two independent sources of randomness from which we extract randomness again and again and preform various computations on the sequence. We denote by $\mathcal{H} = (H_t, \ldots, H_1)$ the (L, R)-history that captures the random variables obtained from L, R so far. Typically we will know that some random variable P is statistically-close to uniform even given $\mathcal{H}$, namely, $(P, \mathcal{H}) \approx (U, \mathcal{H})$. Furthermore, P is either a deterministic function of $L, \mathcal{H}$ or otherwise P is a deterministic function of $R, \mathcal{H}$. Assume, without loss of generality, that P is a deterministic function of $L, \mathcal{H}$. Let Ext be a strong seeded extractor. The following lemma states that if M is a deterministic function of $R, \mathcal{H}$ and $\widetilde{H}_\infty(M \mid \mathcal{H})$ is sufficiently high, then $(\mathsf{Ext}(M, P), P, \mathcal{H}) \approx (U, P, \mathcal{H})$.

The proof of this technical lemma is fairly simple. Nevertheless, we apply the lemma frequently and believe that our proofs are cleaner and conceptually simpler by identifying the operation that is described and analyzed by the lemma as an atomic operation.

Lemma 3.3: Let L, R be two independent random variables, and let $\mathcal{H}$ be an (L, R)-history. Let P be a random variable over $\{0,1\}^p$ which is a deterministic function of $L, \mathcal{H}$. [4] Assume that

$$(P, \mathcal{H}) \approx_\delta (U_p, \mathcal{H}) . \tag{1}$$

Let M be a random variable over $\{0,1\}^m$ which is a deterministic function of $R, \mathcal{H}$, such that

$$\widetilde{H}_\infty (M \mid \mathcal{H}) \geq k + \log(1/\varepsilon). \tag{2}$$

Let $\mathsf{Ext}\colon \{0,1\}^m \times \{0,1\}^p \to \{0,1\}^f$ be a strong seeded extractor for entropy k with error ε. Define $F = \mathsf{Ext}(M, P)$. Then, $P, \mathcal{H}$ is an (L, R)-history, and

$$(F, P, \mathcal{H}) \approx_{\delta+2\varepsilon} (U_f, P, \mathcal{H}) .$$

[4] Note that any (L, R)-history is also an (R, L)-history, and so an analog statement of the lemma in which P is a deterministic function of $R, \mathcal{H}$ readily follows.

The following lemma is also used frequently in our proofs.

Lemma 3.4: Let L, R be two independent random variables, and let $\mathcal{H}$ be an (L, R)-history. Let P be a random variable that is a deterministic function of $R, \mathcal{H}$. Let J be a random variable that is a deterministic function of $L, \mathcal{H}$. Then,

$$\mathsf{SD}\left((P, J, \mathcal{H}), (U, J, \mathcal{H})\right) = \mathsf{SD}\left((P, \mathcal{H}), (U, \mathcal{H})\right).$$

Moreover, $J, \mathcal{H}$ is an (L, R)-history.

IV. Two-Steps Look-Ahead Extractors

In this section we present a restricted version of look-ahead extractors. Building on the idea of alternating extraction [DP07], Dodis and Wichs [DW09] introduced the notion of look-ahead extractors. Look-ahead extractors were further used by Li [Li13a], [Li15] for his multi-source extractors. In these cases, the look-ahead extractors were applied for some non-constant number of "steps" or "rounds". We construct our LCBs using look-ahead extractors with only two steps. This in turn allows us to present relatively simple constructions of LCBs, which are also easier to analyze. Since we need only this very restricted version, and since we use it in the analysis of our constructions in a white-box manner, we give in this section the construction for two-steps look-ahead extractors.

Let n, a, h be integers and let $\varepsilon > 0$ be such that $a = \Omega(\log(h/\varepsilon))$ and $h = \Omega(\log(n/\varepsilon))$. Set $s = \Theta(\log(n/\varepsilon))$, where some appropriately chosen large enough universal constant is hidden under the Θ notation. Let $\mathsf{Ext}_1 \colon \{0,1\}^n \times \{0,1\}^s \to \{0,1\}^a$ and $\mathsf{Ext}_2 \colon \{0,1\}^h \times \{0,1\}^a \to \{0,1\}^s$ be strong seeded extractors from Theorem 2.2, both with error ε. Note that the choice of s and the assumption on a guarantee that the seed lengths of Ext_1 and Ext_2 are sufficient. Moreover, by Theorem 2.2, Ext_1 is an extractor for entropy $2a$ and Ext_2 is an extractor for entropy $2s$. Define the function

$$\mathsf{LookAheadExt} \colon \{0,1\}^h \times \{0,1\}^n \to \{0,1\}^a \times \{0,1\}^a$$

as follows. Given $W \in \{0,1\}^h$ and $Y \in \{0,1\}^n$, let

$$\begin{aligned} A &= \mathsf{Ext}_1(Y, W|_s), \\ Z &= \mathsf{Ext}_2(W, A), \\ B &= \mathsf{Ext}_1(Y, Z). \end{aligned}$$

Define

$$\mathsf{LookAheadExt}(W, Y) = (A, B).$$

With notations as above, we have the following lemma.

Lemma 4.1: Let r be an integer. Let X, Y be two independent random variables, and let $\mathcal{H}$ be an (X, Y)-history such that

$$\widetilde{H}_\infty\left(Y \mid \mathcal{H}\right) \geq (r+2)a + \log(1/\varepsilon). \tag{3}$$

Let W be a random variable of the form of an $r \times h$ matrix, which is a deterministic function of $X, \mathcal{H}$, where

$$h \geq (r+2)s + \log(1/\varepsilon). \tag{4}$$

Assume further that there exists $g \in [r]$ such that

$$(W_g, \mathcal{H}) \approx_\delta (U_h, \mathcal{H}). \tag{5}$$

For each $i \in [r]$, let (A_i, B_i) be the output $\mathsf{LookAheadExt}(W_i, Y)$. Then the following holds:

- $\mathcal{H}' = (W, Z_g, \{A_i\}_{i=1}^r, W|_s, \mathcal{H})$ is an (X, Y)-history.
- $(B_g, \mathcal{H}') \approx_{2\delta+6\varepsilon} (U_a, \mathcal{H}')$.
- $\widetilde{H}_\infty(Y \mid \mathcal{H}') \geq \widetilde{H}_\infty(Y \mid \mathcal{H}) - ra$.

- For any random variable N which is a deterministic function of $X, \mathcal{H}$, it holds that $\widetilde{H}_\infty(N \mid \mathcal{H}') \geq \widetilde{H}_\infty(N \mid \mathcal{H}) - rh$.

As mentioned, Lemma 4.1 is not new and general versions of it appear in the literature. Nevertheless, as we consider a restricted setting and since the lemma as stated uses the notion of (L, R)-histories (which is new), a direct proof for the lemma above does not appear in the literature (though existing proofs can be adopted in a straightforward manner). Thus, for completeness, we give a proof for Lemma 4.1 in the full version of this extended abstract.

V. A Warm Up – Merging Three Rows

In order to convey the ideas underling our LCBs, we present in this section a construction of a merger with weak-seeds for a somewhere-random source with only three rows. This toy example allows us to present some of the ideas used in the actual constructions of our mergers with weak-seeds (Theorem 1.5) and LCBs (Theorem 1.2). This section is meant only for building up intuition, and presenting the underling ideas behind our constructions without getting into all the details.

During this section we ignore the error analysis as this does not affect the parameters and slightly complicates the presentation. In particular, when applying Lemma 3.3 and Lemma 4.1 we ignore the expression $\log(1/\varepsilon)$ in Equation (2) and in Equation (3).

In this section we prove the following theorem which, roughly speaking, states that one can efficiently and deterministically merge the rows of a $3 \times \ell$ somewhere-random source, using an independent (n, k)-weak-source, even when $\ell = \Theta(\log n)$ and $k = \Omega(\log \log n)$.

Theorem 5.1 (Merging three rows): For any integer n, there exists a $\mathrm{poly}(n)$-time computable function

$$\mathsf{Merg}_3 \colon \left(\{0,1\}^\ell\right)^3 \times \{0,1\}^n \to \{0,1\}^m,$$

where $\ell = \Theta(\log n)$ and $m = \Omega(\ell)$, with the following property. Let X be a $3 \times \ell$ somewhere-random source. Let Y be an independent (n, k)-weak-source with $k = \Omega(\log \log n)$. Then, $\mathsf{Merg}_3(X, Y) \approx U_m$.

Proof: During the proof of this toy example we assume that the second row, X_2, is good. Of course, the algorithm Merg_3 will not rely on this assumption (or otherwise the algorithm can simply output X_2). We use the assumption that X_2 is uniform only for the analysis. We exemplify this with the good row being the second row just to avoid introducing more indices. Since the second row is not the first or the last row, this will enable us to demonstrate all the ideas needed to prove the theorem for any number of rows.

We turn to present the construction of Merg_3. For the reader's convenience, the construction is depict in Figure 1. As mentioned, the problem of merging the 3 rows X_1, X_2, X_3, with X_2 being the good row, is reduced to the problem of obtaining random variables X_1''', X_2''', X_3''', where each X_i''' is a function of X_i and Y, with the following property: $(X_2''', X_1''', X_3''') \approx (U, X_1''', X_3''')$, namely, constructing 3-LCBs for 3 rows. Once this independence is obtained, one can simply output

$$\mathsf{Merg}_3(X, Y) = X_1''' \oplus X_2''' \oplus X_3'''.$$

Set h to be just large enough for the two-steps look-ahead extractor from Section IV with $r = 3$. Taking $h = \Theta(\log n)$ will do. Set $a = \Theta(\log \log n)$ and note that this choice of a satisfies the hypothesis of the two-steps look-ahead extractor. We now compute

$$\begin{aligned}
(A_1, B_1) &= \mathsf{LookAheadExt}\left((X_1)|_h, Y\right), \\
(A_2, B_2) &= \mathsf{LookAheadExt}\left((X_2)|_h, Y\right), \\
(A_3, B_3) &= \mathsf{LookAheadExt}\left((X_3)|_h, Y\right),
\end{aligned}$$

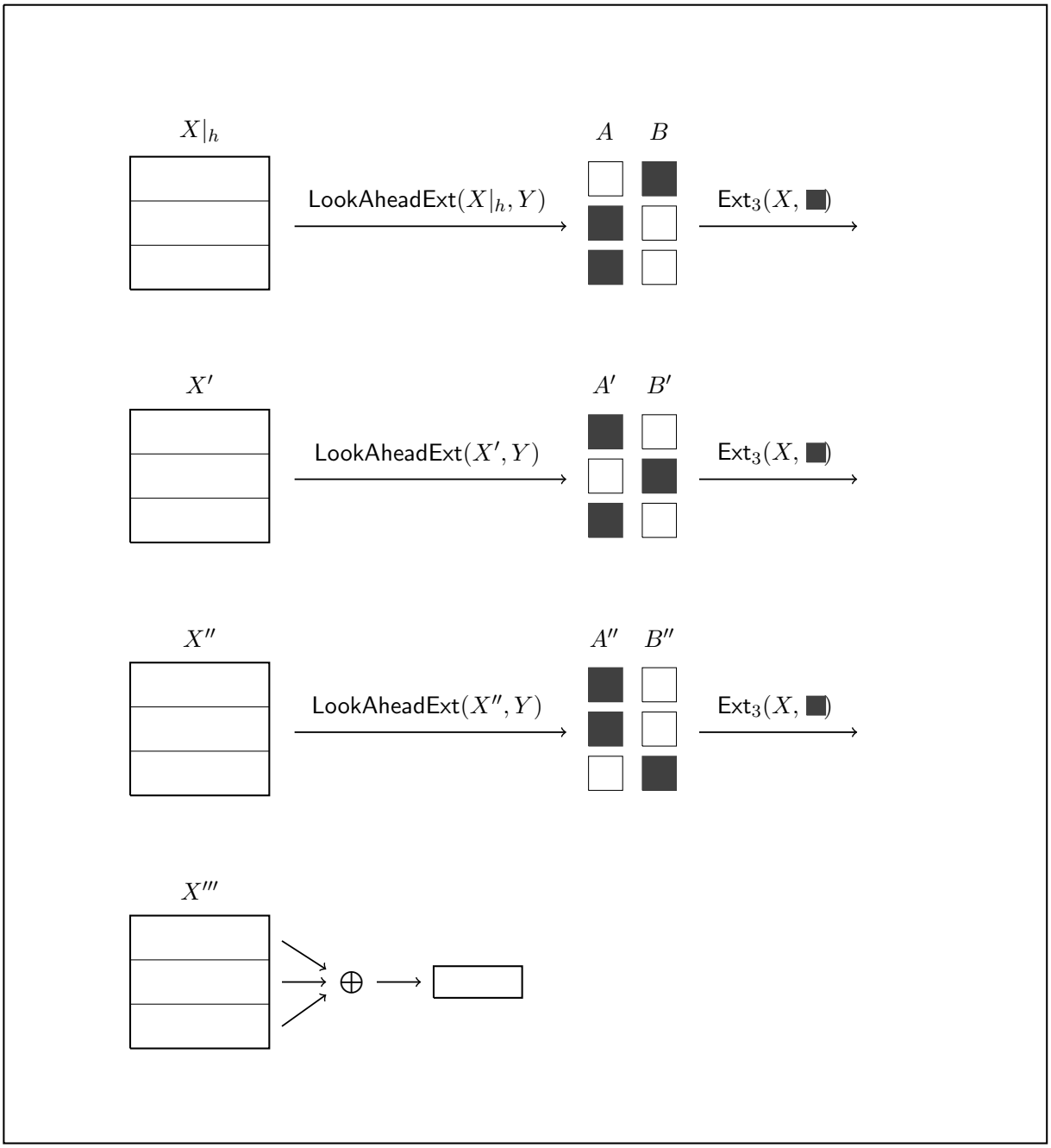

Figure 1. A schematic diagram of the three rows merger Merg_3.

and then compute

$$X'_1 = \mathsf{Ext}_3(X_1, B_1),$$
$$X'_2 = \mathsf{Ext}_3(X_2, A_2),$$
$$X'_3 = \mathsf{Ext}_3(X_3, A_3),$$

where $\mathsf{Ext}_3 \colon \{0,1\}^\ell \times \{0,1\}^a \to \{0,1\}^h$ is the strong seeded extractor from Theorem 2.2 for entropy $2h$. [5] This was the first iteration of the algorithm Merg_3, in which we produced the random variables X'_1, X'_2 and X'_3 from X_1, X_2, X_3 and Y. In the second iteration we will compute X''_1, X''_2 and X''_3 from X'_1, X'_2, X'_3 and X, Y in a similar way. The difference will be that instead of taking the B variable as a seed for the first row and the corresponding A variables to the other rows, we will take the B variable as a seed for the *second* row and the corresponding A variables to the other two rows. More formally, we compute

$$(A'_1, B'_1) = \mathsf{LookAheadExt}\left(X'_1, Y\right),$$
$$(A'_2, B'_2) = \mathsf{LookAheadExt}\left(X'_2, Y\right),$$
$$(A'_3, B'_3) = \mathsf{LookAheadExt}\left(X'_3, Y\right),$$

and then set

$$X''_1 = \mathsf{Ext}_3(X_1, A'_1),$$
$$X''_2 = \mathsf{Ext}_3(X_2, B'_2),$$
$$X''_3 = \mathsf{Ext}_3(X_3, A'_3).$$

The algorithm continues for one more iteration, and computes

$$(A''_1, B''_1) = \mathsf{LookAheadExt}\left(X''_1, Y\right),$$
$$(A''_2, B''_2) = \mathsf{LookAheadExt}\left(X''_2, Y\right),$$
$$(A''_3, B''_3) = \mathsf{LookAheadExt}\left(X''_3, Y\right),$$

and then computes

$$X'''_1 = \mathsf{Ext}_3(X_1, A''_1),$$
$$X'''_2 = \mathsf{Ext}_3(X_2, A''_2),$$
$$X'''_3 = \mathsf{Ext}_3(X_3, B''_3).$$

Informally speaking, we want to show that if there is enough entropy in Y and in X_2 (namely, ℓ is large enough), then X'''_2 is close to uniform even given X'''_1, X'''_3. This is formalized by the following claim. By the discussion above, once we prove Claim 5.2, Theorem 5.1 will follow.

Claim 5.2: If $k \geq 11a$ and $\ell \geq 13h$, then

$$(X'''_2, X'''_1, X'''_3) \approx (U_h, X'''_1, X'''_3).$$

Before proving Claim 5.2, we present the high-level strategy of the proof, which consists of three steps.

- First, we show that in iterations that precede the "good" iteration (that is, the iteration in which the good row is given the B variable, which in our case is the second iteration) the assumption on the input is preserved. Namely, at the end of each such iteration, an output row that corresponds to a good input row is uniform, and the joint distribution of the rows is independent of Y.

[5] We use the name Ext_3 because during the proof we will argue about the random variables obtained by the two-steps look-ahead extractor from Section IV, which uses two strong seeded extractors we denoted by Ext_1 and Ext_2.

- In the second step we show that after the good iteration was executed, the respective output row "gains its independence". That is, an output row that corresponds to a good input row is uniform even conditioned on all other output rows. Moreover, the joint distribution of the rows is independent of Y.
- In the third step we show that the independence of the good row is "preserved" throughout the remaining iterations. Namely, an output row that corresponds to a good input row remains uniform even conditioned on all other output rows, and moreover, the joint distribution of all output rows is independent of Y.

Proof of Claim 5.2: The proof will follow the three iterations of the algorithm. In the first iteration we give the "lead", namely the B variable, to the "wrong" row X_1. We show that nothing bad happens by letting X_1 have the lead, in the following sense: after this iteration, we have that the joint distribution (X'_1, X'_2, X'_3) is independent of Y (more formally, this independence holds conditioned on any fixing of carefully chosen (X, Y)-history), and X'_2 is close to uniform. So besides losing some entropy in Y and in X_2, and observing some error, we maintain the assumption we had initially about our input – the second row X'_2 is uniform, and the joint distribution of the three rows is independent of Y. Thus, in some sense we can "skip" to the iteration in which we give the lead to the good row, which in our case is the second row. This easily generalizes to any number of rows that precede the good row.

Analyzing the first iteration: Recall that $A_2 = \mathsf{Ext}_1(Y, (X_2)|_s)$. Moreover, Y and $(X_2)|_s$ are independent, and $H_\infty(Y) = k \geq 11a$. Since Ext_1 is a strong seeded extractor for entropy $2a$, we have that

$$(A_2, (X_2)|_s) \approx (U_a, \cdot).\ ^6$$

Note further that conditioned on any fixing of $(X_2)|_s$, the random variables A_2 and $X|_h$ are independent. Thus, we can apply Lemma 2.6 and conclude that

$$(A_2, X|_h) \approx (U_a, \cdot). \tag{6}$$

Recall that $X'_2 = \mathsf{Ext}_3(X_2, A_2)$. We apply Lemma 3.3 to the (X, Y)-history $X|_h$ with $P = A_2$, $M = X_2$ and the extractor Ext_3. The hypothesis of Lemma 3.3 is met since A_2 is a deterministic function of Y and $X|_h$. Moreover, since Ext_3 is an extractor for entropy $2h$, and since

$$\widetilde{H}_\infty(X_2 \mid (X|_h)) \geq \ell - 3h \geq 10h,$$

Equation (2) of Lemma 3.3 holds. Therefore, Lemma 3.3 together with Equation (6) imply that

$$(X'_2, A_2, X|_h) \approx (U_h, \cdot).$$

Moreover, $A_2, X|_h$ is an (X, Y)-history.

We now apply Lemma 3.4 with $P = X'_2$, $J = (B_1, A_3)$ and the (X, Y)-history $A_2, X|_h$. Since X'_2 is a deterministic function of X and A_2 and since B_1, A_3 are deterministic functions of Y and $X|_h$, Lemma 3.4 implies that $\mathcal{H}_1 = (B_1, A_2, A_3, X|_h)$ is an (X, Y)-history and that

$$(X'_2, \mathcal{H}_1) \approx (U_h, \cdot). \tag{7}$$

Since each of B_1, A_2, A_3 consists of a bits and since Y is independent of $X|_h$, Lemma 2.11 and Lemma 2.13 imply that

$$\widetilde{H}_\infty(Y \mid \mathcal{H}_1) \geq \widetilde{H}_\infty(Y \mid (X|_h)) - 3a = H_\infty(Y) - 3a \geq 8a. \tag{8}$$

Similarly, conditioned on any fixing of $X|_h$, the random variables B_1, A_2, A_3 are deterministic functions of Y, whereas X_2 is a deterministic function of X. Hence, Lemma 2.13 implies that $\widetilde{H}_\infty(X_2 \mid \mathcal{H}_1) = \widetilde{H}_\infty(X_2 \mid (X|_h))$. Since $X|_h$ consists of $3h$ bits, we have that

$$\widetilde{H}_\infty(X_2 \mid \mathcal{H}_1) \geq H_\infty(X_2) - 3h = 10h. \tag{9}$$

This concludes the first iteration. Note that after the first iteration X'_2 is close to uniform (Equation (7)). Moreover, Y and X_2 still have (enough) entropy (Equation (8), Equation (9)).

[6] Recall that our notation dictates that $(X, Z_1, \ldots, Z_r) \approx (Y, \cdot)$ is a shorthand for $(X, Z_1, \ldots, Z_r) \approx (Y, Z_1, \ldots, Z_r)$

Analyzing the second iteration: We reached the iteration in which we give the lead to the good row – X_2. We want to show that after this iteration, $(X_2'', X_1'',\ X_3'') \approx (U_h, X_1'',\ X_3'')$. Namely, the good row "gains its independence" in the iteration in which it takes the lead.

We continue from Equation (7) and apply Lemma 4.1 to the (X,Y)-history $\mathcal{H}_1$, with the $3 \times h$ matrix X' and the weak-source Y. Equation (3) of Lemma 4.1 holds by Equation (8). Since $h \geq 5s$, Equation (4) of Lemma 4.1 holds as well. Therefore, Lemma 4.1 together with Equation (7) imply that

$$\mathcal{H}_1' = (X', Z_2', A_1', A_2', A_3', (X')|_s, \mathcal{H}_1)$$

is an (X,Y)-history, and that

$$(B_2', \mathcal{H}_1') \approx (U_a, \cdot).$$

Furthermore, the third item of Lemma (4.1) together with Equation (8) imply that

$$\widetilde{H}_\infty\left(Y \mid \mathcal{H}_1'\right) \geq \widetilde{H}_\infty\left(Y \mid \mathcal{H}_1\right) - 3a \geq 5a. \tag{10}$$

The fourth item of Lemma 4.1, applied with $N = X_2$, together with Equation (9), implies that

$$\widetilde{H}_\infty\left(X_2 \mid \mathcal{H}_1'\right) \geq \widetilde{H}_\infty\left(X_2 \mid \mathcal{H}_1\right) - 3h \geq 7h. \tag{11}$$

We now apply Lemma 3.4 to the (X,Y)-history $\mathcal{H}_1'$ with $P = B_2'$ and $J = (X_1'', X_3'')$. Lemma 3.4 is applicable since B_2' is a deterministic function of Y, X_2', and the latter is contained in $\mathcal{H}_1'$. Moreover, X_1'', X_3'' are deterministic functions of X, A_1', A_3', and A_1', A_3' are contained in $\mathcal{H}_1'$. Thus, Lemma 3.4 implies that

$$(B_2', X_1'', X_3'', \mathcal{H}_1') \approx (U_a, \cdot), \tag{12}$$

and that $X_1'', X_3'', \mathcal{H}_1'$ is an (X,Y)-history.

Recall that $X_2'' = \mathsf{Ext}_3(X_2, B_2')$. We now apply Lemma 3.3 to the (X,Y)-history $X_1'', X_3'', \mathcal{H}_1'$ with $P = B_2'$, $M = X_2$ and the extractor Ext_3. The hypothesis of the lemma holds since B_2' is a deterministic function of Y and X_2', and the latter is contained in $\mathcal{H}_1'$. By Equation (11) and Lemma 2.11, it holds that

$$\widetilde{H}_\infty\left(X_2 \mid X_1'', X_3'', \mathcal{H}_1'\right) \geq \widetilde{H}_\infty\left(X_2 \mid \mathcal{H}_1'\right) - 2h \geq 5h. \tag{13}$$

Since Ext_3 is a strong seeded extractor for entropy $2h$, Equation (2) of Lemma 3.3 holds. Therefore, Lemma 3.3 together with Equation (12) imply that

$$(X_2'', \mathcal{H}_2) \approx (U_h, \cdot), \tag{14}$$

where $\mathcal{H}_2 = (B_2', X_1'', X_3'', \mathcal{H}_1')$ is an (X,Y)-history. In terms of entropy-loss,

$$\widetilde{H}_\infty\left(Y \mid \mathcal{H}_2\right) \geq \widetilde{H}_\infty\left(Y \mid X_1'', X_3'', \mathcal{H}_1'\right) - a = \widetilde{H}_\infty\left(Y \mid \mathcal{H}_1'\right) - a \geq 4a, \tag{15}$$

where the first inequality follows by Lemma 2.11 and the fact that $|B_2'| = a$. The second equality follows by Lemma 2.13 and the fact that conditioned on any fixing of $\mathcal{H}_1'$, the random variables X_1'', X_3'' are deterministic functions of X, and are thus independent of Y. The last inequality follows by Equation (10).

Similarly, since B_2' is independent of X_2 conditioned on any fixing of $\mathcal{H}_1'$, we have that

$$\widetilde{H}_\infty\left(X_2 \mid \mathcal{H}_2\right) = \widetilde{H}_\infty\left(X_2 \mid X_1'', X_3'', \mathcal{H}_1'\right) \geq 5h, \tag{16}$$

where the last inequality follows by Equation (13). Since X_1'', X_3'' are contained in $\mathcal{H}_2$, this proves what we wanted for this iteration. Namely, after the second iteration, in which the good row takes the lead, $(X_2'', X_1'', X_3'') \approx (U_h, X_1'', X_3'')$.

Analyzing the third iteration: We now show that the independence of the good row X_2'' is "preserved" throughout the following iteration, where again another row takes the lead. We continue from Equation (14). We note that conditioned on any fixing of $\mathcal{H}_2$, the random variables A_1'', B_3'' are deterministic functions of Y (as X_1'', X_3'' are contained in $\mathcal{H}_2$). On the other hand, conditioned on any fixing of $\mathcal{H}_2$, the random variable X_2'' is a deterministic function of X (as B_2' is contained in $\mathcal{H}_2$). Thus, by Lemma 3.4 applied to the (X,Y)-history $\mathcal{H}_2$ with $P = X_2''$ and $J = (A_1'', B_3'')$, it holds that

$$(X_2'', A_1'', B_3'', \mathcal{H}_2) \approx (U_h, \cdot).$$

Furthermore, $A_1'', B_3'', \mathcal{H}_2$ is an (X,Y)-history. By Lemma 2.11 and Equation (15), it holds that

$$\widetilde{H}_\infty\left(Y \mid A_1'', B_3'', \mathcal{H}_2\right) \geq \widetilde{H}_\infty\left(Y \mid \mathcal{H}_2\right) - 2a \geq 2a. \tag{17}$$

Recall that $A_2'' = \mathsf{Ext}_1(Y, (X_2'')|_s)$. We apply Lemma 3.3 to the (X,Y)-history $A_1'', B_3'', \mathcal{H}_2$ with $P = (X_2'')|_s$, $M = Y$ and the extractor Ext_1. Since Ext_1 is an extractor for entropy $2a$, Equation (2) of Lemma 3.3 holds by Equation (17). Furthermore, $(X_2'')|_s$ is a deterministic function of X and B_2', which is contained in $\mathcal{H}_2$. Thus, the hypothesis of Lemma 3.3 is met, and we get that

$$(A_2'', X_2'', A_1'', B_3'', \mathcal{H}_2) \approx (U_a, \cdot),$$

and $X_2'', A_1'', B_3'', \mathcal{H}_2$ is an (X,Y)-history. In terms of entropy-loss, since $|X_2''| = h$ and since A_1'', B_3'' are deterministic functions of Y conditioned on any fixing of X_1'', X_3'', which are contained in $\mathcal{H}_2$, Lemma 2.11 together with Equation (16) imply that

$$\widetilde{H}_\infty\left(X_2 \mid X_2'', A_1'', B_3'', \mathcal{H}_2\right) \geq \widetilde{H}_\infty\left(X_2 \mid A_1'', B_3'', \mathcal{H}_2\right) - h = \widetilde{H}_\infty\left(X_2 \mid \mathcal{H}_2\right) - h \geq 4h. \tag{18}$$

We now apply Lemma 3.4 to the (X,Y)-history $X_2'', A_1'', B_3'', \mathcal{H}_2$ with $P = A_2''$ and $J = X_1''', X_3'''$. Recall that $X_1''' = \mathsf{Ext}_3(X_1, A_1'')$ and $X_3''' = \mathsf{Ext}_3(X_3, B_3'')$. Thus, conditioned on any fixing of A_1'', B_3'', it holds that X_1''', X_3''' are deterministic functions of X, whereas A_2'' is a deterministic function of Y conditioned on any fixing of X_2''. Thus, Lemma 3.4 implies that

$$(A_2'', X_1''', X_3''', \mathcal{H}_2') \approx (U_a, \cdot),$$

where $\mathcal{H}_2' = (X_2'', A_1'', B_3'', \mathcal{H}_2)$ is an (X,Y)-history. In terms of entropy-loss, by Equation (18) and Lemma 2.11, we have that

$$\widetilde{H}_\infty\left(X_2 \mid X_1''', X_3''', \mathcal{H}_2'\right) \geq \widetilde{H}_\infty\left(X_2 \mid X_2'', A_1'', B_3'', \mathcal{H}_2\right) - 2h \geq 2h. \tag{19}$$

Recall that $X_2''' = \mathsf{Ext}_3(X_2, A_2'')$. We apply Lemma 3.3 to the (X,Y)-history $X_1''', X_3''', \mathcal{H}_2'$, with $P = A_2''$, $M = X_2$ and the extractor Ext_3. Note that A_2'' is a deterministic function of Y and X_2'', which is contained in $\mathcal{H}_2'$. Equation (2) of Lemma 3.3 follows by Equation (19) and the fact that Ext_3 is an extractor for entropy $2h$. Lemma 3.3 then implies that

$$(X_2''', A_2'', X_1''', X_3''', \mathcal{H}_2') \approx (U_h, \cdot).$$

By Lemma 2.7 it follows that

$$(X_2''', X_1''', X_3''') \approx (U_h, \cdot),$$

which concludes the proof of the claim.

■

As mentioned above, the proof of Theorem 5.1 readily follows by Claim 5.2. ■

A. Merging r rows – an overview

Generalizing the proof of the three-rows merger presented above to $r > 3$ rows is straightforward. Instead of three iterations, we can apply the algorithm above for r iterations, where at the i^{th} iteration we give the lead to row i. Working out the parameters, one can show that this generalization works for $\ell = O(r^4 \cdot \log n)$ and $k = O(r^3 \cdot \log\log n)$. We now explain how one can improve this, and construct a merger for $\ell = \tilde{O}(r^2) \cdot \log n$ and $k = \tilde{O}(r) \cdot \log\log n$, as we obtain in the actual construction (see the full version of the paper).

For the purpose of constructing mergers with weak-seeds, this improvement, although desired, is not crucial, especially when r is small. This, for example, is the case in the construction of our three-source extractor. Thus, in these cases, the simpler merger depicted above is sufficient. However, for our construction of LCBs the somewhat more involved construction is necessary, and so in the rest of this section we give an informal overview of the actual construction.

Consider the complete graph on r vertices, where vertex $i \in [r]$ represents the i^{th} row of X. In the straightforward generalization of the three rows merger to r rows, we (implicitly) considered r cuts of this graph, where the i^{th} cut is $(\{i\}, [r] \setminus \{i\})$. The construction in Theorem 5.1 guarantees that if X_i is good then after the i^{th} iteration, row i is uniform even given all other rows (and remains as such throughout the following iterations). In the actual construction of our merger (and LCBs) we generalize this idea and guarantee that the following stronger property holds. For any cut (S, S^c) of $[r]$, the i^{th} row is independent of all rows with indices that are separated from i by the cut (S, S^c). Notice that when we used the cuts of the form $(\{i\}, [r] \setminus \{i\})$, we knew that at some iteration the good row $g \in [r]$ is separated from all other rows, and moreover, we knew on which side of the cut g will be (the side that contains the single vertex). By inspecting the construction above, one can see that the algorithm used this second piece of information. Indeed, in each iteration we gave the lead to the single row, namely, we gave the single row the B variable, and all other rows got the A variables.

When considering general cuts (S, S^c), we no longer know which side of the cut contains the good row g. Namely, to who should we give the B variables – to the rows in S or to the rows in S^c. We solve this problem by applying the construction used above twice, in a "flip-flop". Namely, we first give the B variables to the rows in S and the A variables to the rows in S^c, and then run one more round, giving the B variables to the rows in S^c and the A variables to the rows in S. We only then proceed to the next cut in the sequence.

Having the ability to use general cuts allows us to run for only $\log r$ iterations rather than for r iterations. Indeed, instead of choosing r (highly unbalanced) cuts, and run for r iterations, we use $q = \log r$ (efficiently computable) cuts $S_1, \ldots, S_q$ with the following property. For any two distinct $i, j \in [r]$, there exists $v \in [q]$ such that the cut (S_v, S_v^c) separates i from j. By working with these cuts, the same independence guarantee holds when the algorithm terminates. Indeed, after the v^{th} iteration, row i is uniform and independent of all rows that were separated from i by at least one of the cuts $(S_1, S_1^c), \ldots, (S_v, S_v^c)$. By the property of $S_1, \ldots, S_q$ it follows that after all q iterations were executed, row i is uniform and independent of all other rows. Since we run for only $q = \log r$ iterations, as apposed to r iterations, we obtain a multiplicative saving of roughly $r/\log r$ in both k, ℓ, which yields the desired improvement.

Our construction of LCBs follows the same idea as the construction of the mergers described above. The only difference is that the analysis is done "locally" on t rows, rather than on r rows. The fact that we run for $\log r$ iterations introduces only logarithmic factors of r into k, ℓ, as apposed to polynomial factors.

REFERENCES

[BKS+05] B. Barak, G. Kindler, R. Shaltiel, B. Sudakov, and A. Wigderson. Simulating independence: New constructions of condensers, Ramsey graphs, dispersers, and extractors. In *Proceedings of the thirty-seventh annual ACM Symposium on Theory of Computing*, pages 1–10. ACM, 2005.

[Bou05] J. Bourgain. More on the sum-product phenomenon in prime fields and its applications. *International Journal of Number Theory*, 1(01):1–32, 2005.

[BRSW12] B. Barak, A. Rao, R. Shaltiel, and A. Wigderson. 2-source dispersers for $n^{o(1)}$ entropy, and Ramsey graphs beating the Frankl-Wilson construction. *Annals of Mathematics*, 176(3):1483–1544, 2012.

[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.

[CGL15] E. Chattopadhyay, V. Goyal, and X. Li. Non-malleable extractors and codes, with their many tampered extensions. *arXiv preprint arXiv:1505.00107*, 2015.

[CRS14] G. Cohen, R. Raz, and G. Segev. Nonmalleable extractors with short seeds and applications to privacy amplification. *SIAM Journal on Computing*, 43(2):450–476, 2014.

[CZ15] E. Chattopadhyay and D. Zuckerman. Explicit two-source extractors and resilient functions. *Electronic Colloquium on Computational Complexity (ECCC)*, 2015.

[DKSS09] Z. Dvir, S. Kopparty, S. Saraf, and M. Sudan. Extensions to the method of multiplicities, with applications to Kakeya sets and mergers. In *50th Annual IEEE Symposium on Foundations of Computer Science*, pages 181–190. IEEE, 2009.

[DLWZ14] Y. Dodis, X. Li, T. D. Wooley, and D. Zuckerman. Privacy amplification and nonmalleable extractors via character sums. *SIAM Journal on Computing*, 43(2):800–830, 2014.

[DORS08] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM Journal on Computing*, 38(1):97–139, 2008.

[DP07] S. Dziembowski and K. Pietrzak. Intrusion-resilient secret sharing. In *48th Annual IEEE Symposium on Foundations of Computer Science*, pages 227–237, 2007.

[DR08] Z. Dvir and R. Raz. Analyzing linear mergers. *Random Structures & Algorithms*, 32(3):334–345, 2008.

[DS07] Z. Dvir and A. Shpilka. An improved analysis of linear mergers. *computational complexity*, 16(1):34–59, 2007.

[DW09] Y. Dodis and D. Wichs. Non-malleable extractors and symmetric key cryptography from weak secrets. In *Proceedings of the forty-first annual ACM Symposium on Theory of Computing*, pages 601–610. ACM, 2009.

[DW11] Z. Dvir and A. Wigderson. Kakeya sets, new mergers, and old extractors. *SIAM Journal on Computing*, 40(3):778–792, 2011.

[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.

[Li11] X. Li. Improved constructions of three source extractors. In *IEEE 26th Annual Conference on Computational Complexity*, pages 126–136, 2011.

[Li12a] X. Li. Design extractors, non-malleable condensers and privacy amplification. In *Proceedings of the forty-fourth annual ACM Symposium on Theory of Computing*, pages 837–854, 2012.

[Li12b] X. Li. Non-malleable condensers for arbitrary min-entropy, and almost optimal protocols for privacy amplification. *arXiv preprint arXiv:1211.0651*, 2012.

[Li13a] X. Li. Extractors for a constant number of independent sources with polylogarithmic min-entropy. In *IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 100–109, 2013.

[Li13b] X. Li. New independent source extractors with exponential improvement. In *Proceedings of the forty-fifth annual ACM Symposium on Theory of Computing*, pages 783–792. ACM, 2013.

[Li15] X. Li. Three-source extractors for polylogarithmic min-entropy. *Electronic Colloquium on Computational Complexity (ECCC)*, 2015.

[LRVW03] C.J. Lu, O. Reingold, S. Vadhan, and A. Wigderson. Extractors: Optimal up to constant factors. In *Proceedings of the thirty-fifth annual ACM Symposium on Theory of Computing*, pages 602–611. ACM, 2003.

[Rao09] A. Rao. Extractors for a constant number of polynomially small min-entropy independent sources. *SIAM Journal on Computing*, 39(1):168–194, 2009.

[Raz05] R. Raz. Extractors with weak random seeds. In *Proceedings of the thirty-seventh annual ACM Symposium on Theory of Computing*, pages 11–20, 2005.

[TS96] A. Ta-Shma. On extracting randomness from weak random sources. In *Proceedings of the twenty-eighth annual ACM Symposium on Theory of Computing*, pages 276–285, 1996.

[Zuc07] D. Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. *Theory of Computing*, 3:103–128, 2007.