Two-Source and Affine Non-Malleable Extractors for Small Entropy

Xin Li
Johns Hopkins University, Baltimore, MD, USA

Yan Zhong
Johns Hopkins University, Baltimore, MD, USA

Abstract

Non-malleable extractors are generalizations and strengthening of standard randomness extractors, that are resilient to adversarial tampering. Such extractors have wide applications in cryptography and have become important cornerstones in recent breakthroughs of explicit constructions of two-source extractors and affine extractors for small entropy. However, explicit constructions of non-malleable extractors appear to be much harder than standard extractors. Indeed, in the well-studied models of two-source and affine non-malleable extractors, the previous best constructions only work for entropy rate $> 2/3$ and $1 - \gamma$ for some small constant $\gamma > 0$ respectively by Li (FOCS' 23).

In this paper, we present explicit constructions of two-source and affine non-malleable extractors that match the state-of-the-art constructions of standard ones for small entropy. Our main results include:

- Two-source and affine non-malleable extractors (over F_2) for sources on n bits with min-entropy $k \geq \log^C n$ and polynomially small error, matching the parameters of standard extractors by Chattopadhyay and Zuckerman (STOC' 16, Annals of Mathematics' 19) and Li (FOCS' 16).
- Two-source and affine non-malleable extractors (over F_2) for sources on n bits with min-entropy $k = O(\log n)$ and constant error, matching the parameters of standard extractors by Li (FOCS' 23).

Our constructions significantly improve previous results, and the parameters (entropy requirement and error) are the best possible without first improving the constructions of standard extractors. In addition, our improved affine non-malleable extractors give strong lower bounds for a certain kind of read-once linear branching programs, recently introduced by Gryaznov, Pudlák, and Talebanfard (CCC' 22) as a generalization of several well studied computational models. These bounds match the previously best-known average-case hardness results given by Chattopadhyay and Liao (CCC' 23) and Li (FOCS' 23), where the branching program size lower bounds are close to optimal, but the explicit functions we use here are different. Our results also suggest a possible deeper connection between non-malleable extractors and standard ones.

2012 ACM Subject Classification Theory of computation → Pseudorandomness and derandomization

Keywords and phrases Randomness Extractors, Non-malleable, Two-source, Affine

Digital Object Identifier 10.4230/LIPIcs.ICALP.2024.108

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: `https://arxiv.org/abs/2404.17013`

Funding *Xin Li*: Supported by NSF CAREER Award CCF-1845349 and NSF Award CCF-2127575. *Yan Zhong*: Supported by NSF CAREER Award CCF-1845349.

1 Introduction

Randomness extractors are fundamental objects in the broad area of pseudorandomness. These objects have been studied extensively and found applications in diverse areas such as cryptography, complexity theory, combinatorics and graph theory, and many more.

Informally, randomness extractors are functions that transform imperfect randomness called weak random sources into almost uniform random bits. Originally, the motivation for studying these objects comes from the gap between the requirement of high-quality random bits in various computational and cryptographic applications, and the severe biases in natural random sources. In practice, weak random sources can arise in several different situations. For example, the random bits can become biased and correlated due to the natural process that generates them, or because of the fact that an adversary learns some partial information about a random string in cryptographic applications.

To measure the amount of randomness in a weak random source (a random variable) X, we use the standard definition of *min-entropy*: $H_\infty(X) = \min_{x \in \mathsf{supp}(X)} \log_2(1/\mathbf{Pr}[X = x])$. If $X \in \{0,1\}^n$, we say X is an $(n, H_\infty(X))$-source, or simply an $H_\infty(X)$-source if n is clear from context. We also say X has *entropy rate* $H_\infty(X)/n$. Ideally, one would like to construct deterministic extractors for all (n, k) sources when k is not too small. However, this is well known to be impossible, even if one only desires to extract one bit and k is as large as $n - 1$. Thus, to allow randomness extraction one has to put additional restrictions on the source.

Historically, many different models of randomness extractors have been studied. For example, if one gives the extractor an additional independent short uniform random seed, then there exist extractors that work for any (n, k) source. Such extractors, first introduced by Nisan and Zuckerman [64], are known as *seeded extractors*. These extractors have found wide applications, and by now we have almost optimal constructions (e.g., [62, 42, 35, 34]) after a long line of research.

However, seeded extractors may not be applicable in situations where the short uniform random seed is either not available (e.g., in cryptography) or cannot be simulated by cycling over all possible choices. For these applications, one needs *deterministic extractors* or *seedless extractors*, and many different models have also been studied in this setting. These include for example extractors for independent sources [20, 2, 3, 67, 7, 65, 4, 49, 52, 54, 53, 55, 22, 26, 18, 56, 30, 12, 23, 5, 27, 28, 57, 59, 47, 60], bit fixing sources [21, 46, 39, 66], affine sources [38, 8, 66, 72, 6, 69, 50, 56, 10, 60], samplable sources [70, 71], interleaved sources [68, 18], and small-space sources [45]. We define deterministic extractors below.

▶ **Definition 1.** *Let $\mathcal{X}$ be a family of distribution over $\{0,1\}^n$. A function $\mathsf{Ext} : \{0,1\}^n \to \{0,1\}^m$ is a deterministic extractor for $\mathcal{X}$ with error ε if for every distribution $X \in \mathcal{X}$, we have*

$$\mathsf{Ext}(X) \approx_\varepsilon U_m,$$

where U_m stands for the uniform distribution over $\{0,1\}^m$, and $\approx_\varepsilon$ means ε-close in statistical distance. We say Ext is explicit if it is computable by a polynomial-time algorithm.

Among these models, two of the most well-studied are extractors for independent sources and affine sources. This is in part due to their connections to several other areas of interest. For example, extractors for independent sources are useful in distributed computing and cryptography with imperfect randomness [44, 43], and give explicit constructions of Ramsey graphs; while affine sources generalize bit-fixing sources, and extractors for affine sources have applications in exposure-resilient cryptography [21, 46] as well as Boolean circuit lower bounds [31, 37, 48].

Using simple probabilistic arguments, one can show that there exist extractors for two independent (n, k) sources with $k = \log n + O(1)$, which is optimal up to the constant $O(1)$. The first explicit construction of two-source extractors was given by Chor and Goldreich [20], which achieves $k > n/2$. Following a long line of research and several recent breakthroughs,

we now have explicit constructions of two-source extractors for entropy $k \approx 4n/9$ with error $\varepsilon = 2^{-\Omega(n)}$ [47], for entropy $k = \text{polylog}(n)$ with error $\varepsilon = 1/\mathsf{poly}(n)$ [18], and for entropy $k = O(\log n)$ with constant error [60]. Similarly, for affine sources which are uniform distributions over some unknown affine subspace over the vector space F_2^n, [1] one can show the existence of extractors for entropy $k = O(\log n)$, which is also optimal up to the constant $O(1)$. Regarding explicit constructions, we have affine extractors for entropy $k = \delta n$ with error $\varepsilon = 2^{-\Omega(n)}$ for any constant $\delta > 0$ [8, 72, 50], for entropy $k = \text{polylog}(n)$ with error $\varepsilon = 1/\mathsf{poly}(n)$ [56], and for entropy $k = O(\log n)$ with constant error [60].

In the past decade or so, a new kind of extractors, known as *non-malleable extractors*, has gained a lot of attention. These extractors are motivated from cryptographic applications. Informally, the setting is that an adversary can tamper with the inputs to an extractor in some way, and the non-malleable extractor guarantees that the output of the extractor is close to uniform even conditioned on the output of the extractor on the tampered inputs. The most well-studied non-malleable extractors include seeded non-malleable extractors [33], two-source non-malleable extractors [19], and affine non-malleable extractors [15]. These non-malleable extractors have wide applications in cryptography, such as privacy amplification with an active adversary [33] and non-malleable codes [36]. Furthermore, they turn out to have surprising connections to the constructions of standard extractors. Indeed, starting from the work of Li [52] which showed a connection between seeded non-malleable extractors and two-source extractors, these non-malleable extractors have played key roles, and now become important cornerstones in the recent series of breakthroughs that eventually lead to explicit constructions of two-source and affine extractors for asymptotically optimal entropy. In a more recent line of work [41, 17, 61], a special case of affine non-malleable extractors known as *directional affine extractors* is also shown to give strong lower bounds for certain read-once branching programs with linear queries, which generalize both standard read-once branching programs and parity decision trees. Given these applications, non-malleable extractors have become important objects that deserve to be studied on their own. We now define tampering functions and two kinds of non-malleable extractors below.

▶ **Definition 2** (Tampering Function). *For any function $f : S \to S$, We say f has no fixed points if $f(s) \neq s$ for all $s \in S$. For any $n > 0$, let $\mathcal{F}_n$ denote the set of all functions $f : \{0,1\}^n \to \{0,1\}^n$. Any subset of $\mathcal{F}_n$ is a family of tampering functions.*

▶ **Definition 3** ([19]). *A function $\mathsf{2nmExt} : (\{0,1\}^n)^2 \to \{0,1\}^m$ is a (k_1, k_2, ε) two source non-malleable extractor, if it satisfies the following property: Let X, Y be two independent, (n, k_1) and (n, k_2) sources, and $f, g : \{0,1\}^n \to \{0,1\}^n$ be two arbitrary tampering functions such that at least one of them has no fixed point,[2] then*

$$|\mathsf{2nmExt}(X, Y) \circ \mathsf{2nmExt}(f(X), g(Y)) - U_m \circ \mathsf{2nmExt}(f(X), g(Y))| < \varepsilon.$$

▶ **Definition 4** ([15]). *A function $\mathsf{anmExt} : \{0,1\}^n \to \{0,1\}^m$ is a (k, ε) affine non-malleable extractor if for any affine source X with entropy at least k and any affine function $f : \{0,1\}^n \to \{0,1\}^n$ with no fixed point, we have*

$$|\mathsf{anmExt}(X) \circ \mathsf{anmExt}(f(X)) - U_m \circ \mathsf{anmExt}(f(X))| \leq \varepsilon.$$

[1] In this paper we focus on the case where the field is F_2, for larger fields there are affine extractors with better parameters.

[2] We say that x is a fixed point of a function f if $f(x) = x$.

Using the probabilistic method, one can also prove the existence of these non-malleable extractors with excellent parameters. For example, [19] showed that two-source non-malleable extractors exist for (n, k) sources when $k \geq m + \frac{3}{2}\log(1/\varepsilon) + O(1)$ and $k \geq \log n + O(1)$. Similarly, it can be also shown that affine non-malleable extractors exist for entropy $k \geq 2m + 2\log(1/\varepsilon) + \log n + O(1)$.

However, constructing explicit non-malleable extractors appears to be significantly harder than constructing standard extractors, despite considerable effort. Indeed, even for seeded non-malleable extractors, the initial explicit constructions [32, 29, 51] only work for sources with entropy rate $> 1/2$, and it was not until [11] that explicit seeded non-malleable extractors for sources with poly-logarithmic entropy are constructed. After a long line of research [32, 29, 51, 52, 11, 24, 25, 12, 15, 23, 27, 28, 57, 59, 60], an asymptotically optimal seeded non-malleable extractor is finally constructed in [60]. On the other hand, the situation for two-source non-malleable extractors and affine non-malleable extractors is much worse, where the best-known constructions in [60] only achieve entropy $k > 2n/3$ and $k \geq (1-\gamma)n$ for a small constant $\gamma > 0$. This is in sharp contrast to the constructions of standard two-source and affine extractors, where explicit constructions can work for entropy $k = \text{polylog}(n)$ with polynomially small error [18, 56], and for entropy $k = O(\log n)$ with constant error [60].

1.1 Our Results

In this paper, we study two-source and affine non-malleable extractors for small entropy. Our main results give explicit constructions of such non-malleable extractors that essentially match their standard counterparts in the small entropy regime. Specifically, we give explicit two-source and affine non-malleable extractors for $\text{polylog}(n)$ entropy with polynomially small error and for $O(\log n)$ entropy with constant error. We have the following theorems.

▶ **Theorem 5.** *There exists a constant $C > 1$ such that for any $k \geq \log^C n$, there exists an explicit construction of a $(k, k, n^{-\Omega(1)})$ two-source non-malleable extractor with output length $\Omega(k)$.*

▶ **Theorem 6.** *There exists a constant $C > 1$ such that for any $k \geq \log^C n$, there exists an explicit construction of a $(k, n^{-\Omega(1)})$ affine non-malleable extractor with output length $k^{\Omega(1)}$.*

▶ **Theorem 7.** *There exists a constant $c > 1$ such that for any $k \geq c\log n$, there exists an explicit construction of a $(k, k, O(1))$ two-source non-malleable extractor with output length 1.*

▶ **Theorem 8.** *There exists a constant $c > 1$ such that for any $k \geq c\log n$, there exists an explicit construction of a $(k, O(1))$ affine non-malleable extractor with output length 1.*

▶ Remark 9. The output length in the two theorems for entropy $k \geq c\log n$ can be extended to a constant number by using the standard XOR lemma and previous techniques (e.g., those in [56]). Furthermore, our constructions can also be extended to handle multiple tampering functions as in [11]. For simplicity, we omit the details here.

The following tables summarize our results compared to some of the best previous constructions.

Table 1 Prior and current results on two-source non-malleable extractors.

Two-source Non-malleable Extractor	Entropy k_1	Entropy k_2	Output m	Error ε
[11]	$n - n^{\gamma}$	$n - n^{\gamma}$	$n^{\Omega(1)}$	$2^{-n^{\Omega(1)}}$
[59]	$(1-\gamma)n$	$(1-\gamma)n$	$\Omega(n)$	$2^{-\Omega(\frac{n \log\log n}{\log n})}$
[1]	$(\frac{4}{5}+\delta)n$	$\log^C n$	$\Omega(\min\{k_1,k_2\})$	$2^{-\min\{k_1,k_2\}^{\Omega(1)}}$
[60]	$(\frac{2}{3}+\gamma)n$	$k = O(\log n)$	$\Omega(k)$	$2^{-\Omega(k)}$
This work (Theorem 5)	$k \geq \text{polylog}(n)$	$k \geq \text{polylog}(n)$	$\Omega(k)$	$n^{-\Omega(1)}$
This work (Theorem 7)	$O(\log n)$	$O(\log n)$	1	$O(1)$

Table 2 Prior and current results on affine non-malleable extractors.

Affine Non-malleable Extractor	Entropy k	Output m	Error ε
[15]	$n - n^{\delta}$ for some constant $\delta \in (0,1)$	$n^{\Omega(1)}$	$2^{-n^{\Omega(1)}}$
[60]	$(1-\gamma)n$, $\gamma < 1/1000$	$\Omega(n)$	$2^{-\Omega(n)}$
This work (Theorem 6)	$\text{polylog}(n)$	$k^{\Omega(1)}$	$n^{-\Omega(1)}$
This work (Theorem 8)	$O(\log n)$	1	$O(1)$

Our results thus significantly improve the entropy requirement of previous non-malleable extractors. As a comparison, we list below the best-known explicit two-source extractors and affine extractors for small entropy.

Table 3 Best-known results on two-source extractors.

Two-source Extractor	Entropy k	Output m	Error ε
[18]	$\text{polylog}(n)$	1	$n^{-\Omega(1)}$
[63, 56, 13]	$\text{polylog}(n)$	$k^{\Omega(1)}$	$n^{-\Omega(1)}$
[5]	$O(\log n 2^{O(\sqrt{\log\log n})})$	1	$O(1)$
[27]	$O(\log n (\log\log n)^{O(1)})$	1	$O(1)$
[58]	$O(\log n \log\log n)$	1	$O(1)$
[59]	$O(\log n \frac{\log\log n}{\log\log\log n})$	1	$O(1)$
[60]	$O(\log n)$	1	$O(1)$

Table 4 Best-known results on affine extractors.

Affine Extractor	Entropy k	Output m	Error ε
[56]	$\text{polylog}(n)$	$k^{\Omega(1)}$	$n^{-\Omega(1)}$
[10]	$O(\log n \log\log n \log\log\log^6 n)$	1	$O(1)$
[16]	$O(\log n \log\log n \log\log\log^3 n)$	1	$O(1)$
[60]	$O(\log n)$	1	$O(1)$

It can be seen that the parameters of our two-source and affine non-malleable extractors essentially match those of standard two-source and affine extractors for small entropy. We also point out that the error of our non-malleable extractors is the best one can hope for without first improving the error of standard two-source and affine extractors for small entropy, since the non-malleable extractors are stronger versions of extractors, and in particular, they are themselves two-source and affine extractors. Finally, given that our constructions use many of the key components in the constructions of standard extractors, we believe that any future

techniques that improve the error of standard two-source and affine extractors for small entropy (e.g., to negligible error) are also likely applicable to our constructions to get the same improvement on the error of two-source and affine non-malleable extractors.

1.2 Applications to Lower bounds for Read-Once Linear Branching Programs

Our affine non-malleable extractors have applications in proving average-case hardness against read-once linear branching programs (ROLBPs). This computational model was recently introduced by Gryaznov, Pudlák, and Talebanfard [41] as a generalization of several important and well-studied computational models such as decision trees, parity decision trees, and standard read-once branching programs. Roughly, a read-once linear branching program is a branching program that can make linear queries to the input string, while these queries are linearly independent along any path. Formally, we have the following definition.

▶ **Definition 10** (Linear branching program [41]). *A linear branching program on F_2^n is a directed acyclic graph P with the following properties:*

- *There is only one source s in P.*
- *There are two sinks in P, labeled with 0 and 1 respectively.*
- *Every non-sink node v is labeled with a linear function $\ell_v : \mathsf{F}_2^n \to \mathsf{F}_2$. Moreover, there are exactly two outgoing edges from v, one is labeled with 1 and the other is labeled with 0.*

The size of P is the number of non-sink nodes in P. P computes a Boolean function $f : \{0,1\}^n \to \{0,1\}$ in the following way. For every input $x \in \mathsf{F}_2^n$, P follows the computation path by starting from s, and when on a non-sink node v, moves to the next node following the edge with label $\ell_v(x) \in \{0,1\}$. The computation ends when the path ends at a sink, and $f(x)$ is defined to be the label on this sink.

[41] defines two kinds of read-once linear branching programs (ROLBP for short).

▶ **Definition 11** ([41]). *Given any linear branching program P and any node v in P, let Pre_v denote the span of all linear queries that appear on any path from the source to v, excluding the query ℓ_v. Let Post_v denote the span of all linear queries in the subprogram starting at v.*

- *A linear branching program P is weakly read-once if for every inner node v of P, it holds that $\ell_v \notin \mathsf{Pre}_v$.*
- *A linear branching program P is strongly read-once if for every inner node v of P, it holds that $\mathsf{Pre}_v \cap \mathsf{Post}_v = \{0\}$.*

Both kinds of ROLBPs generalize the aforementioned computational models, but weakly read-once linear branching programs (WROLBPs) are more flexible than strongly read-once linear branching programs (SROLBPs). As a result, proving lower bounds for WROLBPs turns out to be much harder than for SROLBPs. Indeed, so far we only have non-trivial lower bounds for SROLBPs. To state our results, we use the following definition.

▶ **Definition 12** ([17]). *For a Boolean function $f : \{0,1\}^n \to \{0,1\}$, let $\mathsf{SROLBP}(f)$ denote the smallest possible size of a strongly read-once linear branching program that computes f, and $\mathsf{SROLBP}_\varepsilon(f)$ denote the smallest possible size of a strongly read-once linear branching program P such that*

$$\mathbf{Pr}_{x \leftarrow_U \mathsf{F}_2^n}[P(x) = f(X)] \geq \frac{1}{2} + \varepsilon.$$

The definition can be adapted to ROBPs naturally.

[41] shows that a stronger version of affine extractors known as *directional affine extractors* give strong average case lower bounds for SROLBPs. They give an explicit construction of directional affine extractors for entropy $k \geq \frac{2n}{3} + c$ with error $\varepsilon \leq 2^{-c}$ for any constant $c > 1$, which also implies exponential average-case hardness for SROLBPs of size up to $2^{\frac{n}{3} - o(n)}$. In a follow-up work, Chattopadhyay and Liao [17] used another kind of extractors known as *sumset extractors* [14] to give an alternative average-case hardness for SROLBPs. In particular, they gave an explicit function Ext such that $\mathsf{SROLBP}_{n^{-\Omega(1)}}(\mathsf{Ext}) \geq 2^{n - \log^{O(1)} n}$. More recently, Li [60] gave an improved sumset extractor which in turn yields an explicit function Ext such that $\mathsf{SROLBP}_{2^{-\Omega(1)}}(\mathsf{Ext}) \geq 2^{n - O(\log n)}$. In these two constructions, the branching program size lower bounds become quite close to optimal (the result of [60] is optimal up to the constant in $O(\cdot)$), while the correlation becomes polynomially large or a constant. Another recent work by Li and Zhong [61] gave explicit directional affine extractor for entropy $k \geq cn(\log\log\log n)^2/\log\log n$ with error $\varepsilon = 2^{-n^{\Omega(1)}}$ for some constant $c > 1$, which implies exponential average-case hardness for SROLBPs of size up to $2^{n - o(n)}$.

For simplicity, we do not define directional affine extractors here, but just mention that directional affine extractors are a special case of affine non-malleable extractors. Hence, our new constructions of affine non-malleable extractors directly imply improved directional affine extractors, which in turn also give average-case hardness for SROLBPs. Specifically, we have the following theorem.

▶ **Theorem 13.** *There exist explicit functions* anmExt_1, anmExt_2 *such that* $\mathsf{SROLBP}_{n^{-\Omega(1)}}(\mathsf{anmExt}_1) \geq 2^{n - \log^{O(1)} n}$ *and* $\mathsf{SROLBP}_{2^{-\Omega(1)}}(\mathsf{anmExt}_2) \geq 2^{n - O(\log n)}$.

These bounds match the previously best-known average-case hardness results for SROLBPs given in [17] and [60], where the branching program size lower bounds are close to optimal, but the explicit functions we use here are different. Specifically, here we use affine non-malleable extractors while [17] and [60] use sumset extractors.

2 Technical Overview

Here we outline the main techniques used in this paper, opting for an informal approach at times for clarity while omitting certain technical details.

We use the standard notation in the literature where a letter with $'$ represents a tampered version. Let f and g denote the tampering functions on X and Y in two-source non-malleable extractors, respectively, and $\mathcal{A}$ be the affine tampering function in affine non-malleable extractors.

Since two-source and affine non-malleable extractors are themselves two-source and affine extractors, our high-level idea is to adapt the constructions of standard extractors for polylogarithmic or logarithmic entropy into the stronger, non-malleable version. Clearly, a direct naive application of standard extractors may not work, since the output on the tampered inputs may be correlated to the output on the original inputs. Below we start with two-source extractors to illustrate our main ideas. Let us first briefly review the constructions of two-source extractors for small entropy. Generally, these extractors are double-layered: the outer layer is a suitable resilient function, which is designed to be an extractor for non-oblivious bit-fixing (NOBF) sources with t-wise independent property for some parameter t. That is, most of the bits are t-wise independently uniform, while the rest of the bits can depend arbitrarily on these bits. Here, the extractor uses a crucial property that bounded independence suffices to work for several resilient functions (or equivalently these functions are *fooled* by bounded independence), such as the derandomized Ajtai-Linial function in [18]

or the Majority function. The inner layer is a transformation that transforms two independent sources into a single NOBF source with the t-wise independent property. This step itself utilizes techniques from seeded non-malleable extractors or correlation breakers, which are functions designed to break correlations between random variables.

To adapt the construction to two-source non-malleable extractors, our first observation is that there is an easy case. Intuitively, this is the case where one input source has large entropy conditioned on the tampered version. For instance, say the source X has high entropy conditioned on every fixing of $X' = f(X) = x'$. Then in the analysis, we can first fix X', and then further fix the tampered output of the extractor, which is now a deterministic function of Y and can be chosen to have a relatively small size. Conditioned on these fixings, we have X and Y are still independent and have good entropy, hence any two-source extractor will give an output that is close to uniform conditioned on the tampered output.

However, it is certainly possible that the above does not hold. For example, the tampering function f can be an injective function, so that conditioned on any fixing of $X' = f(X) = x'$, we have that X is also fixed. In this case, our observation is that X' itself must also have large entropy (since f injective), therefore we can possibly create structures in the distribution of the tampered version as well. Specifically, our strategy is to modify the inner layer of the two-source extractor while essentially using the same outer layer. For simplicity, let us consider extractors with just one bit of output. A standard approach to show the output bit is close to uniform conditioned on the tampered output, is to show that the parity of these two bits is close to uniform. Since the outer extractor is a resilient function, this suggests to look at the parity of two copies of resilient functions on two correlated distributions.

Now another crucial observation behind our construction is that just like in the construction of standard two-source extractors, for certain resilient functions, the parity of two copies of such functions is still fooled by bounded independence. Thus, if in the inner layer, we can create structures such that the *joint distribution* of the NOBF source and the tampered version has the t-wise independent property, then we will be able to show that the extractor is non-malleable. Note that we are now in the case where the tampered sources also have high entropy, which works in our favor since achieving t-wise independence requires a certain amount of entropy. However, we cannot simply use previous techniques since the tampered sources are correlated with the original sources. Therefore, we appropriately modify previous constructions of correlation breakers to ensure the t-wise independent property in the joint distribution.

Finally, in the actual analysis, we are not guaranteed to be in either case; and it may happen that for some x', conditioned on $X' = x'$ we have that X has large entropy, while for others conditioned on $X' = x'$ we have that X has small entropy. The analysis thus needs a careful interpolation between different cases in terms of a convex combination of subsources. We now elaborate with more details on each of these aspects below.

First we give some notation that will help with our presentation.

▶ **Definition 14** (t-non-malleable (k, ε) seeded extractor)**.** *A function* $\mathsf{nmExt} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *is a t-non-malleable (k, ε) extractor if it satisfies the following property: if X is a (n, k)-source and Y is uniform on $\{0,1\}^d$, and $f_1, \cdots, f_t$ are arbitrary functions from d bits to d bits with no fixed point, then*

$$(\mathsf{nmExt}(X, Y), \mathsf{nmExt}(X, f_1(Y)), \cdots, \mathsf{nmExt}(X, f_t(Y)), Y) \approx_\varepsilon (U_m, \mathsf{nmExt}(X, f_1(Y)), \cdots, Y).$$

We say a distribution or a source X on n bits is (q, t, γ) independent if there exists a subset $S \subseteq [n]$ with $|S| \leq q$ such that if we consider the bits of X in $[n] \setminus S$, then every t bits are γ-close to uniform.

A t-non-malleable (k, ε) seeded extractor nmExt with seed length d can be used to generate a (q, t, γ) source from a source X with entropy at least k in the following way: cycle over all possible seeds i, and for each one output a bit $\mathsf{nmExt}(X, i)$. The output is now $(\sqrt{\varepsilon}D, t+1, t\sqrt{\varepsilon})$ independent with $D = 2^d$.

2.1 Taking the parity of two resilient functions

A Boolean function on n variables is a resilient function if it is nearly balanced, and no small coalition can have a significant influence on the output of the function. Such functions are equivalent to extractors for NOBF sources. The resilient functions that have played a key role in the recent advancement of extractors are the derandomized Ajtai-Linial function in [18] and the Majority function. The former is a monotone AC^0 function, that is fooled by polylog(n)-wise independence, while the latter is a threshold function that can be fooled by constant-wise independence.

It is not hard to show that the parity of two independent copies of resilient functions is still a resilient function. What is left to show is that such a parity can also be fooled by bounded independence. When the resilient function is in AC^0, we observe that the parity of two such functions is also in AC^0, because the parity of two bits can be written as a constant size AC^0 circuit. Therefore, the parity of two derandomized Ajtai-Linial functions is still in AC^0, and can be fooled by polylog(n)-wise independence by Braverman's celebrated result [9] on bounded independence fooling AC^0 circuits, together with some standard techniques.

To show that constant-wise independence fools the parity of two Majority functions, we use the work of Gopalan, O'Donnell, Wu, and Zuckerman [40], which shows that constant-wise independence fools any function of halfspaces under product distributions, as long as the function can be implemented as a constant size circuit. In our case, this clearly holds since we are just taking the parity of two Majority functions. Using the XOR lemma and previous techniques (e.g., those in [56], our construction can also be extended to output a constant number of bits.

2.2 Generating NOBF sources from the inputs and its tampered counterparts

We want to construct a function such that when the tampered sources have sufficient entropy, the joint distribution of the generated bits from the input sources and the tampered sources is (q, t, γ) independent for some suitable parameters q, t, and γ.

The standard approach for two-source extractors, as introduced in [18], is to first apply a seeded non-malleable extractor to one source, say Y, and then use another source X to sample a small number of bits from the output. However, in our case, this black-box approach does not work since the tampered sources are correlated with the original inputs. Therefore, we have to create some kind of difference between the tampered sources and the original sources, which will enable us to get the desired (q, t, γ) independent property.

To achieve this, we dig into the constructions of seeded non-malleable extractors and existing two-source non-malleable extractors, which roughly go as follows. First, one uses an *advice generator* to create a short string that is different from the tampered version with high probability. Then, conditioned on the fixing of the advice strings, one can argue that the two sources are still independent and have sufficient entropy. At this point one uses a *correlation breaker with advice*, together with the advice strings to compute the output, which is guaranteed to have the non-malleable property. However, the steps of generating advice and subsequent application of correlation breakers require the sources to have very large entropy (e.g., at least 2/3), which is the main reason that previous two-source non-malleable extractors can only work for large entropy.

To get around this barrier, our approach is to first apply a standard seeded extractor to one source Y, and output say $\Omega(k)$ bits where k is the entropy. By cycling over all possible seeds, we potentially get a matrix with $D = 2^d$ rows where d is the seed length of the seeded extractor. We then use the other source X to sample a small number ($\mathsf{poly}(n)$) of rows from the output. Now, again a standard argument implies that most of the rows are close to uniform. Since we are in the case where the tampered sources X', Y' have sufficient entropy, this is also true for the tampered version. Note that we haven't achieved the (q, t, γ) independent property yet. Our next step is to generate advice from the original sources and the tampered sources. However, in the low-entropy regime, it is hard to generate a single advice for the input sources and the tampered version – the advice generator requires generating uniform seeds to sample from an encoding of the inputs and it is hard to do so from a slice of the sources which could have zero entropy. Therefore, we generate advice from each row. We can then append the index of this row to the advice. This ensures that the advice strings are both different from the tampered version, and also different between different rows. Now, we can apply existing constructions of correlation breakers with advice, which will ensure that for any t rows in the combined matrix from the original sources and the tampered sources, as long as all these rows have high entropy initially, the joint distribution of the final outputs from the correlation breaker is γ-close to uniform.

However, there are additional tricky issues with this approach. First, the correlation breaker requires two independent sources to work, while in our case the outputs in the matrices are already functions of both X and Y. Second, the analysis of the correlation breaker usually requires fixing the advice strings first and arguing that the sources still have sufficient entropy, but now since the matrices have $\mathsf{poly}(n)$ rows and the entropy of the sources is just $k = \mathrm{polylog}(n)$, or even $k = O(\log n)$, if we fix all the advice strings then conditioned on the fixing the sources may not have any entropy left. Finally, the set of "good" rows (the rows that are close to uniform after the sampling using X) in the matrices depends on the source X and Y, and after we fix the advice strings in the analysis, X and Y may have become different, and this could potentially change the set of "good" rows in the first place.

To solve these issues, we use an argument similar to that in [55]. The idea is that since eventually we only need (q, t, γ) independence, in the analysis we can just focus on every subset of t rows from the good rows. In particular, we can set t and the entropy k appropriately, i.e., t is relatively small compared to k. This is because we only need $t = \mathrm{polylog}(n)$ to apply the derandomized Ajtai-Linial in [18] and $t = O(1)$ to apply the Majority function. Now in the analysis, notice that the process of sampling using the source X basically corresponds to $\mathsf{Ext}(Y, \mathsf{Ext}'(X, i))$ where $\mathsf{Ext}, \mathsf{Ext}'$ are two seeded extractors. Thus when t is small, for any subset T with $|T| = t$, we can first fix all $\mathsf{Ext}'(X, i)$ with $i \in T$. By restricting the size of $\mathsf{Ext}'(X, i)$, X still has sufficient entropy conditioned on these fixings, and now the t rows of the outputs $\mathsf{Ext}(Y, \mathsf{Ext}'(X, i))$ are deterministic functions of Y, while X and Y are still independent. By restricting the size of the advice strings, we can preserve the above properties when the analysis fixes the advice strings and goes into the correlation breaker. Finally, as in the analysis in [55], the final error pays a price of a $\mathsf{poly}(n)^t$ factor from a union bound on all possible subsets of size t, which is still fine as long as we set $k \gg t \log n$ and use seeded extractors with error $2^{-\Omega(k)}$ in the correlation breaker.

2.3 Convex combination of subsources

In the above two subsections, we dealt with the case where the tampered sources have sufficient entropy. We now sketch the analysis for the general case.

Let 2nmExt be the two-source non-malleable extractor which works if both X and X' have entropy at least k_x, and both Y and Y' have entropy at least k_y, with error $\varepsilon/2$ and output length m. Now assume X has min-entropy $2k_x + \log(2/\varepsilon)$, and Y has min-entropy $2k_y + \log(2/\varepsilon)$. Further assume without loss of generality that both X and Y are flat sources, i.e., uniform distributions over some unknown subset. The analysis goes by considering the "heavy" elements in the tampered sources X' and Y'. Specifically, for any $x' \in \{0,1\}^n$ and $y' \in \{0,1\}^n$, we consider the pre-image size of $X' = x'$ and $Y' = y'$. If one of them is large, say without loss of generality that the pre-image size of $X' = x'$ is at least 2^{k_x}, then $H_\infty(X|X' = x) \geq k_x$. We can first fix $X' = x'$ and then $\mathsf{2nmExt}(x', Y')$, which is a deterministic function of Y now conditioned on the fixing of $X' = x'$. Since $\mathsf{2nmExt}(x', Y')$ is short compared to $H_\infty(Y)$, conditioned on these fixings we have that X and Y are still independent and have sufficient entropy, so $\mathsf{2nmExt}(X, Y)$ is close to uniform because 2nmExt is itself a two-source extractor. Note that we have already fixed $\mathsf{2nmExt}(x', Y')$, and thus 2nmExt is indeed non-malleable in this case.

Next, consider the set of all the x' whose pre-image size under f is at most 2^{k_x}, and call it BAD_X. If the total probability mass of these x' is at most $\varepsilon/2$, then we can just ignore them (and the corresponding x in the support of X) since this only adds an extra error of $\varepsilon/2$. Similarly, we can also ignore the set of all the y' whose pre-image size under g is at most 2^{k_y} (call it BAD_Y), if the total probability mass of these y' is at most $\varepsilon/2$. In either case, we are done. Otherwise, the subsource of X' formed by all the $x' \in \mathrm{BAD}_X$ has min-entropy at least $-\log(2^{k_x}/(\varepsilon 2^{2k_x}/\varepsilon)) = k_x$, and the corresponding subsource of X has min-entropy at least $2k_x$. Similarly, the subsource of Y' formed by all the $y' \in \mathrm{BAD}_Y$ has min-entropy at least k_y, and the corresponding subsource of Y has min-entropy at least $2k_y$. In this case, both sources and their tampered versions have sufficient entropy, thus by the analysis before, 2nmExt is also a non-malleable extractor.

Since X is just a convex combination of subsources $(X \mid X' = x' \in \mathrm{BAD}_x)$ and $\{(X \mid X' = x' \notin \mathrm{BAD}_x)\}$, and the same is true for Y, the correctness of 2nmExt follows.

Finally, we note that we can modify the two-source non-malleable extractor to output $k^{\Omega(1)}$ bits, by using a similar approach based on the XOR lemma as in [56]. Then, since the two-source non-malleable extractor is strong, we can further apply a standard seeded extractor to increase the output length to $\Omega(k)$.

2.4 Affine non-malleable extractors

Our construction of affine non-malleable extractors roughly follows the same ideas. The difference is that now we do not have access to two independent sources, but the affine source itself has nice structures and the tampering function is affine. Thus, by using appropriate linear seeded extractors as in previous works, certain parts of the affine source behave like independent sources. Therefore, we can suitably adapt our construction of two-source non-malleable extractors to affine non-malleable extractors. One particularly nice property of affine sources is that when applying a strong linear seeded extractor on an affine source, the output on most seeds *is uniform*. This implies that we can generate from an affine source a somewhere random source with no error. In the two-source case, we cannot analyze the generation of NOBF source directly from the definition of the correlation breaker (and have to resort to additional techniques as mentioned in previous paragraph 2.2) due to the error of the somewhere random source. In the affine case, there is no such concern. Therefore, we can argue that we obtain a NOBF source directly from the definition of affine correlation breaker, as in prior works on affine extractors for small entropy (e.g., [10]).

3 Conclusion and Open Problems

In this paper, we significantly improved constructions of two-source and affine non-malleable extractors, and our constructions essentially match standard extractors in the regime of small entropy. We note that any future improvement of extractors for NOBF sources (e.g., improvement in the error) can also translate into improvements of our two-source and affine non-malleable extractors. Furthermore, our results suggest that there may be a deeper connection between standard extractors and their non-malleable counterparts, since their constructions and parameters appear quite similar. In particular, previous works have extensively used non-malleable extractors to construct standard extractors, but is it possible that the reverse direction may also be true? That is, can one also use standard extractors to construct non-malleable extractors?

References

1 Divesh Aggarwal, Eldon Chung, and Maciej Obremski. Extractors: Low entropy requirements colliding with non-malleability. In *Advances in Cryptology – CRYPTO 2023: 43rd Annual International Cryptology Conference, CRYPTO 2023, Santa Barbara, CA, USA, August 20–24, 2023, Proceedings, Part II*, pages 580–610, Berlin, Heidelberg, 2023. Springer-Verlag.

2 Boaz Barak, R. Impagliazzo, and Avi Wigderson. Extracting randomness using few independent sources. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 384–393, 2004.

3 Boaz Barak, Guy Kindler, Ronen Shaltiel, Benny Sudakov, and Avi Wigderson. Simulating independence: New constructions of condensers, Ramsey graphs, dispersers, and extractors. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 1–10, 2005.

4 Boaz Barak, Anup Rao, Ronen Shaltiel, and Avi Wigderson. 2 source dispersers for $n^{o(1)}$ entropy and Ramsey graphs beating the Frankl-Wilson construction. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006.

5 Avraham Ben-Aroya, Dean Doron, and Amnon Ta-Shma. An efficient reduction from two-source to non-malleable extractors: Achieving near-logarithmic min-entropy. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, pages 1185–1194, New York, NY, USA, 2017. Association for Computing Machinery.

6 Eli Ben-Sasson and Swastik Kopparty. Affine dispersers from subspace polynomials. *SIAM J. Comput.*, 41(4):880–914, 2012. doi:10.1137/110826254.

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

8 Jean Bourgain. On the construction of affine extractors. *GAFA Geometric And Functional Analysis*, 17:33–57, January 2007. doi:10.1007/s00039-007-0593-z.

9 Mark Braverman. Polylogarithmic independence fools ac0 circuits. *Journal of the ACM*, 57(5), 2010.

10 Eshan Chattopadhyay, Jesse Goodman, and Jyun-Jie Liao. Affine extractors for almost logarithmic entropy. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*, pages 622–633. IEEE, 2021.

11 Eshan Chattopadhyay, Vipul Goyal, and Xin Li. Non-malleable extractors and codes, with their many tampered extensions. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '16, pages 285–298, New York, NY, USA, 2016. Association for Computing Machinery.

12 Eshan Chattopadhyay and Xin Li. Explicit non-malleable extractors, multi-source extractors and almost optimal privacy amplification protocols. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science*, 2016.

13 Eshan Chattopadhyay and Xin Li. Explicit non-malleable extractors, multi-source extractors, and almost optimal privacy amplification protocols. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 158–167, 2016. doi:10.1109/FOCS.2016.25.

14 Eshan Chattopadhyay and Xin Li. Extractors for sumset sources. In Daniel Wichs and Yishay Mansour, editors, *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC, Cambridge, MA, USA, June 18-21, 2016*, pages 299–311. ACM, 2016. doi:10.1145/2897518.2897643.

15 Eshan Chattopadhyay and Xin Li. Non-malleable codes and extractors for small-depth circuits, and affine functions. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 1171–1184. ACM, 2017.

16 Eshan Chattopadhyay and Jyun-Jie Liao. Extractors for sum of two sources. In Stefano Leonardi and Anupam Gupta, editors, *STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022*, pages 1584–1597. ACM, 2022. doi:10.1145/3519935.3519963.

17 Eshan Chattopadhyay and Jyun-Jie Liao. Hardness against linear branching programs and more. In *Proceedings of the Conference on Proceedings of the 38th Computational Complexity Conference*, CCC '23, Dagstuhl, DEU, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

18 Eshan Chattopadhyay and David Zuckerman. Explicit two-source extractors and resilient functions. *Annals of Mathematics*, 189(3):653–705, 2019.

19 Mahdi Cheraghchi and Venkatesan Guruswami. Non-malleable coding against bit-wise and split-state tampering. In *TCC*, pages 440–464, 2014.

20 Benny Chor and Oded Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. *SIAM Journal on Computing*, 17(2):230–261, 1988.

21 Benny Chor, Oded Goldreich, Johan Hastad, Joel Friedman, Steven Rudich, and Roman Smolensky. The bit extraction problem of t-resilient functions (preliminary version). In *26th Annual Symposium on Foundations of Computer Science, Portland, Oregon, USA, 21-23 October 1985*, pages 396–407, 1985.

22 Gil Cohen. Local correlation breakers and applications to three-source extractors and mergers. In *Proceedings of the 56th Annual IEEE Symposium on Foundations of Computer Science*, 2015.

23 Gil Cohen. Making the most of advice: New correlation breakers and their applications. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science*, 2016.

24 Gil Cohen. Non-malleable extractors - new tools and improved constructions. In *Proceedings of the 31st Annual IEEE Conference on Computational Complexity*, 2016.

25 Gil Cohen. Non-malleable extractors with logarithmic seeds. Technical Report TR16-030, ECCC, 2016.

26 Gil Cohen. Two-source dispersers for polylogarithmic entropy and improved ramsey graphs. In Daniel Wichs and Yishay Mansour, editors, *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 278–284. ACM, 2016. doi:10.1145/2897518.2897530.

27 Gil Cohen. Two-source extractors for quasi-logarithmic min-entropy and improved privacy amplification protocols. Technical Report TR16-114, ECCC: Electronic Colloquium on Computational Complexity, 2016.

28 Gil Cohen. Towards optimal two-source extractors and ramsey graphs. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 1157–1170. ACM, 2017. doi:10.1145/3055399.3055429.

29 Gil Cohen, Ran Raz, and Gil Segev. Non-malleable extractors with short seeds and applications to privacy amplification. *SIAM Journal on Computing*, 43(2):450–476, 2014.

30 Gil Cohen and Leonard Schulman. Extractors for near logarithmic min-entropy. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science*, 2016.

31 Evgeny Demenkov and Alexander Kulikov. An elementary proof of 3n-o(n) lower bound on the circuit complexity of affine dispersers. In *Proceedings of the 36th international conference on Mathematical foundations of computer science*, pages 256–265, 2011.

32 Yevgeniy Dodis, Xin Li, Trevor D. Wooley, and David Zuckerman. Privacy amplification and nonmalleable extractors via character sums. *SIAM Journal on Computing*, 43(2):800–830, 2014.

33 Yevgeniy Dodis and Daniel Wichs. Non-malleable extractors and symmetric key cryptography from weak secrets. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, pages 601–610, 2009.

34 Zeev Dvir, Swastik Kopparty, Shubhangi Saraf, and Madhu Sudan. Extensions to the method of multiplicities, with applications to kakeya sets and mergers. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science*, 2009.

35 Zeev Dvir and Avi Wigderson. Kakeya sets, new mergers and old extractors. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, 2008.

36 Stefan Dziembowski, Krzysztof Pietrzak, and Daniel Wichs. Non-malleable codes. In *ICS*, pages 434–452, 2010.

37 Magnus Gausdal Find, Alexander Golovnev, Edward A. Hirsch, and Alexander S. Kulikov. A better-than-3n lower bound for the circuit complexity of an explicit function. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 89–98, 2016. doi:10.1109/FOCS.2016.19.

38 Ariel Gabizon and Ran Raz. Deterministic extractors for affine sources over large fields. *Combinatorica*, 28(4):415–440, 2008. doi:10.1007/s00493-008-2259-3.

39 Ariel Gabizon, Ran Raz, and Ronen Shaltiel. Deterministic extractors for bit-fixing sources by obtaining an independent seed. *SIAM J. Comput.*, 36(4):1072–1094, 2006. doi:10.1137/S0097539705447049.

40 Parikshit Gopalan, Ryan O'Donnell, Yi Wu, and David Zuckerman. Fooling functions of halfspaces under product distributions. In *2010 IEEE 25th Annual Conference on Computational Complexity*, pages 223–234, 2010. doi:10.1109/CCC.2010.29.

41 Svyatoslav Gryaznov, Pavel Pudlák, and Navid Talebanfard. Linear Branching Programs and Directional Affine Extractors. In *37th Computational Complexity Conference (CCC 2022)*, volume 234, pages 4:1–4:16, 2022.

42 V. Guruswami, C. Umans, and S. Vadhan. Unbalanced expanders and randomness extractors from Parvaresh-Vardy codes. *Journal of the ACM*, 56:1–34, 2009.

43 Yael Kalai, Xin Li, and Anup Rao. 2-source extractors under computational assumptions and cryptography with defective randomness. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 617–628, 2009.

44 Yael Tauman Kalai, Xin Li, Anup Rao, and David Zuckerman. Network extractor protocols. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 654–663, 2008.

45 Jesse Kamp, Anup Rao, Salil P. Vadhan, and David Zuckerman. Deterministic extractors for small-space sources. *Journal of Computer and System Sciences*, 77:191–220, 2011. doi:10.1016/j.jcss.2010.06.014.

46 Jesse Kamp and David Zuckerman. Deterministic Extractors for Bit-Fixing Sources and Exposure-Resilient Cryptography. *Siam Journal on Computing*, 36:1231–1247, 2007. doi:10.1137/S0097539705446846.

47 Mark Lewko. An explicit two-source extractor with min-entropy rate near 4/9. *Mathematika*, 65(4):950–957, 2019. doi:10.1112/S0025579319000238.

48 Jiatu Li and Tianqi Yang. 3.1n - o(n) circuit lower bounds for explicit functions. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2022, pages 1180–1193, New York, NY, USA, 2022. Association for Computing Machinery.

49 Xin Li. Improved constructions of three source extractors. In *Proceedings of the 26th Annual IEEE Conference on Computational Complexity*, pages 126–136, 2011.

50 Xin Li. A new approach to affine extractors and dispersers. In *Proceedings of the 26th Annual IEEE Conference on Computational Complexity, CCC*, 2011.

51 Xin Li. Design extractors, non-malleable condensers and privacy amplification. In *Proceedings of the 44th Annual ACM Symposium on Theory of Computing*, pages 837–854, 2012.

52 Xin Li. Non-malleable extractors, two-source extractors and privacy amplification. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, 2012.

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

54 Xin Li. New independent source extractors with exponential improvement. In *Proceedings of the 45th Annual ACM Symposium on Theory of Computing*, pages 783–792, 2013.

55 Xin Li. Three-source extractors for polylogarithmic min-entropy. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 863–882, Los Alamitos, CA, USA, October 2015. IEEE Computer Society.

56 Xin Li. Improved two-source extractors, and affine extractors for polylogarithmic entropy. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 168–177. IEEE Computer Society, 2016.

57 Xin Li. Improved non-malleable extractors, non-malleable codes and independent source extractors. In *Proceedings of the 49th Annual ACM Symposium on Theory of Computing*, 2017.

58 Xin Li. Improved non-malleable extractors, non-malleable codes and independent source extractors. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, pages 1144–1156, New York, NY, USA, 2017. Association for Computing Machinery.

59 Xin Li. Non-malleable extractors and non-malleable codes: Partially optimal constructions. In Amir Shpilka, editor, *34th Computational Complexity Conference, CCC 2019, July 18-20, 2019, New Brunswick, NJ, USA*, volume 137 of *LIPIcs*, pages 28:1–28:49. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.CCC.2019.28.

60 Xin Li. Two source extractors for asymptotically optimal entropy, and (many) more. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, 2023.

61 Xin Li and Yan Zhong. Explicit directional affine extractors and improved hardness for linear branching programs. Technical report, Arxiv, 2023. arXiv:2304.11495.

62 C. J. Lu, Omer Reingold, Salil Vadhan, and Avi Wigderson. Extractors: Optimal up to constant factors. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pages 602–611, 2003.

63 Raghu Meka. Explicit resilient functions matching ajtai-linial. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '17, pages 1132–1148, USA, 2017. Society for Industrial and Applied Mathematics.

64 Noam Nisan and David Zuckerman. Randomness is linear in space. *Journal of Computer and System Sciences*, 52(1):43–52, 1996.

65 Anup Rao. Extractors for a constant number of polynomially small min-entropy independent sources. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006.

66 Anup Rao. Extractors for low-weight affine sources. In *Proc. of the 24th CCC*, 2009.

67 Ran Raz. Extractors with weak random seeds. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 11–20, 2005.

68 Ran Raz and Amir Yehudayoff. Multilinear formulas, maximal-partition discrepancy and mixed-sources extractors. *Journal of Computer and System Sciences*, 77:167–190, 2011. doi:10.1016/j.jcss.2010.06.013.

69 Ronen Shaltiel. Dispersers for affine sources with sub-polynomial entropy. In *Proceedings of the 52nd Annual IEEE Symposium on Foundations of Computer Science*, 2011.

70 Luca Trevisan and Salil P. Vadhan. Extracting Randomness from Samplable Distributions. In *IEEE Symposium on Foundations of Computer Science*, pages 32–42, 2000. doi:10.1109/SFCS.2000.892063.

71 Emanuele Viola. Extractors for circuit sources. *SIAM Journal on Computing*, 43(2):655–672, 2014.

72 Amir Yehudayoff. Affine extractors over prime fields. *Combinatorica*, 31(2):245–256, 2011.