Extractors for Sum of Two Sources

Eshan Chattopadhyay*
eshan@cs.cornell.edu
Cornell University
Ithaca, New York, USA

Jyun-Jie Liao*
jjliao@cs.cornell.edu
Cornell University
Ithaca, New York, USA

ABSTRACT

We consider the problem of extracting randomness from *sumset sources*, a general class of weak sources introduced by Chattopadhyay and Li (STOC, 2016). An (n, k, C)-sumset source $\mathbf{X}$ is a distribution on $\{0, 1\}^n$ of the form $\mathbf{X}_1 + \mathbf{X}_2 + \ldots + \mathbf{X}_C$, where $\mathbf{X}_i$'s are independent sources on n bits with min-entropy at least k. Prior extractors either required the number of sources C to be a large constant or the min-entropy k to be at least $0.51n$.

As our main result, we construct an explicit extractor for sumset sources in the setting of $C = 2$ for min-entropy $\mathrm{poly}(\log n)$ and polynomially small error. We can further improve the min-entropy requirement to $(\log n) \cdot (\log \log n)^{1+o(1)}$ at the expense of worse error parameter of our extractor. We find applications of our sumset extractor for extracting randomness from other well-studied models of weak sources such as affine sources, small-space sources, and interleaved sources.

Interestingly, it is unknown if a random function is an extractor for sumset sources. We use techniques from additive combinatorics to show that it is a disperser, and further prove that an affine extractor works for an interesting subclass of sumset sources which informally corresponds to the "low doubling" case (i.e., the support of $\mathbf{X}_1 + \mathbf{X}_2$ is not much larger than 2^k).

CCS CONCEPTS

• **Theory of computation → Expander graphs and randomness extractors**.

KEYWORDS

extractors, sumsets, small-space sources, correlation breakers

ACM Reference Format:
Eshan Chattopadhyay and Jyun-Jie Liao. 2022. Extractors for Sum of Two Sources. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing (STOC '22), June 20–24, 2022, Rome, Italy*. ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3519935.3519963

1 INTRODUCTION

Randomness is a powerful resource in computer science, and has been widely used in areas such as algorithm design, cryptography, distributed computing, etc. Most of the applications assume the access to perfect randomness, i.e. a stream of uniform and independent random bits. However, natural sources of randomness often generate biased and correlated random bits, and in cryptographic applications there are many scenarios where the adversary learns some information about the random bits we use. This motivates the area of randomness extraction, which aims to construct *randomness extractors*, which are deterministic algorithms that can convert an imperfect random source into a uniform random string.

Formally, the amount of randomness in an imperfect random source $\mathbf{X}$ is captured by its *min-entropy*, defined as $\mathrm{H}_\infty(\mathbf{X}) = \min_{x \in \mathrm{Supp}(\mathbf{X})}(-\log(\Pr[\mathbf{X} = x]))$.[1] We call $\mathbf{X} \in \{0, 1\}^n$ a (n, k)-source if it satisfies $\mathrm{H}_\infty(\mathbf{X}) \geq k$. Ideally we want a deterministic function Ext with entropy requirement $k \ll n$, i.e. for every (n, k)-source $\mathbf{X}$ the output $\mathrm{Ext}(\mathbf{X})$ is close to a uniform string. Unfortunately, a folklore result shows that it is impossible to construct such a function even when $k = n - 1$.

To bypass the impossibility result, researchers have explored two different approaches. The first one is based on the notion of *seeded extraction*, introduced by Nisan and Zuckerman [38]. This approach assumes that the extractor has access to a short independent uniform random seed, and the extractor needs to convert the given source $\mathbf{X}$ into a uniform string with high probability over the seed. Through a successful line of research we now have seeded extractors with almost optimal parameters [23, 25, 35]. In this paper, we focus on the second approach, called *deterministic extraction*, which assumes some structure in the given source. Formally, a deterministic extractor is defined as follows.

Definition 1.1. *Let $\mathcal{X}$ be a family of distribution over $\{0, 1\}^n$. We say a deterministic function* $\mathrm{Ext} : \{0, 1\}^n \to \{0, 1\}^m$ *is a deterministic extractor for $\mathcal{X}$ with error ε if for every distribution $\mathbf{X} \in \mathcal{X}$,*

$$\mathrm{Ext}(\mathbf{X}) \approx_\varepsilon \mathbf{U}_m.$$

We say Ext *is explicit if* Ext *is computable by a polynomial-time algorithm.*

The most well-studied deterministic extractors are multi-source extractors, which assume that the extractor is given C independent (n, k)-sources $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_C$. This model was first introduced by Chor and Goldreich [14]. They constructed explicit two-source extractors with error $2^{-\Omega(n)}$ for entropy $0.51n$, and proved that there exists a two-source extractor for entropy $k = O(\log(n))$ with error $2^{-\Omega(k)}$. Significant progress was made by Chattopadhyay and Zuckerman [13], who showed how to construct an extractor for two sources with entropy $k = \mathrm{polylog}(n)$, after a long line of successful work on independent source extractors (see the references in [13]). The output length was later improved to $\Omega(k)$ by Li [32]. Furthermore, Ben-Aroya, Doron and Ta-Shma [3] showed how to improve the entropy requirement to $O(\log^{1+o(1)}(n))$ for constant

[1] $\mathrm{Supp}(\mathbf{X})$ denotes the support of $\mathbf{X}$. We use log to denote the base-2 logarithm in the rest of this paper.

*Supported by NSF CAREER award 2045576.

STOC '22, June 20–24, 2022, Rome, Italy

ACM ISBN 978-1-4503-9264-8/22/06…$15.00
https://doi.org/10.1145/3519935.3519963

error and 1-bit output. The entropy requirement was further improved in subsequent works [19, 33], and the state-of-the-art result is by Li [34], which requires $k = O(\log(n) \cdot \frac{\log\log(n)}{\log\log\log(n)})$. For a more elaborate discussion, see the survey by Chattopadhyay [4].

Apart from independent sources, many other classes of sources have been studied for deterministic extraction. We briefly introduce some of these research directions here. A well-studied class is oblivious bit-fixing sources [15, 24, 27, 39], where some unknown coordinates are uniform and independent (and the remaining coordinates are fixed). Extractors for such sources have found applications in cryptography [15, 27]. A natural generalization of bit-fixing sources is the class of affine sources, which are uniform distributions over some affine subspaces and have been widely studied in literature (see [6] and references therein). Another important line of work focuses on the class of samplable sources, which are sources sampled by a "low-complexity procedure" such as efficient algorithms [43], small-space algorithms [26] or constant-depth circuits [44]. Researchers have also studied interleaved sources [9, 10, 12, 40], which is a generalization of independent sources such that the bits from different independent sources are permuted in an unknown order.

In this paper, we consider a very general class of sources called *sumset sources*, which was first studied by Chattopadhyay and Li [9]. A sumset source is the sum (XOR) of multiple independent sources, which we formally define as follows.

Definition 1.2. *A source* $\mathbf{X}$ *is a* (n, k, C)-sumset source *if there exist* C *independent* (n, k)*-sources* $\{\mathbf{X}_i\}_{i\in[C]}$ *such that* $\mathbf{X} = \sum_{i=1}^{C} \mathbf{X}_i$. *If the parameters* n, k *are clear from the context, we simply say* X *is a* C*-sumset source.*

Chattopadhyay and Li [9] showed that the class of sumset sources generalize many different classes we mentioned above, including oblivious bit-fixing sources, independent sources, affine sources and small-space sources. They constructed an explicit extractor for (n, k, C)-sumset sources where $k = \mathrm{polylog}(n)$ and C is a large enough constant, and used the extractor to obtain improved extractors for small-space sources and interleaved sources. It is left as an open question in [9] to obtain explicit extractors for small C, and with the most interesting question being whether it is possible to construct an explicit extractor for 2-sumset sources with low min-entropy.

Note that the model of weak sources, that is the sum of two independent sources, captures and generalizes two central settings in seedless extraction: (i) 2-independent sources setting: given access to independent sources $\mathbf{X}_1$ and $\mathbf{X}_2$, clearly $\mathbf{X} = \mathbf{X}_1 + \mathbf{X}_2$ is a 2-sumset source (ii) affine source setting: an affine source $\mathbf{X}$ with min-entropy k can be written as the sum of two independent affine sources $\mathbf{X}_1, \mathbf{X}_2$, each with min-entropy k.[2] Thus, an extractor for the sum of two sources directly gives a *two-source extractor* as well as an *affine extractor*. As we discuss in Section 1.1, such an extractor yields further improved extractors for interleaved sources and small-space sources as well.

However, it has been challenging to construct extractors for 2-sumset sources with low min-entropy. The only known extractor for the sum of two sources before this work is the Paley graph extractor [14], which requires one source to have entropy $0.51n$ and the other to have entropy $O(\log(n))$, based on character sum estimates obtained by Karatsuba [28, 29] (see also [12, Theorem 4.2]). In fact, unlike other sources we discussed above, it is not clear whether a random function is an extractor for sumset sources. (See Section 1.3 for more discussion.)

In this paper, we give a positive answer to the question above. Formally, we prove the following theorem.

Theorem 1. *There exists a universal constant* C *such that for every* $k \geq \log^C(n)$*, there exists an explicit extractor* $\mathrm{Ext} : \{0,1\}^n \to \{0,1\}^m$ *for* $(n, k, 2)$*-sumset source with error* $n^{-\Omega(1)}$ *and output length* $m = k^{\Omega(1)}$.

We can further lower the entropy requirement to almost logarithmic at the expense of worse error parameter of the extractor.

Theorem 2. *For every constant* $\varepsilon > 0$*, there exists a constant* C_ε *such that there exists an explicit extractor* $\mathrm{Ext} : \{0,1\}^n \to \{0,1\}$ *with error* ε *for* $(n, k, 2)$*-sumset source where*

$$k = C_\varepsilon \log(n) \log\log(n) \log\log\log^3(n).$$

As we noted above, a sumset source extractor is also an affine extractor, and hence Theorem 2 also gives an affine extractor with entropy $O(\log(n)\log\log(n)\log\log\log^3(n))$, which slightly improves upon the $O(\log(n)\log\log(n)\log\log\log^6(n))$ bound in [6]. This improvement comes from a new construction of an "affine correlation breaker", which we discuss in Section 1.2.

1.1 Applications

Next we discuss some applications of our sumset extractors in obtaining improved extractors for other well-studied models of weak sources.

1.1.1 Extractors for Interleaved Sources. Interleaved sources are a natural generalization of independent sources, introduced by Raz and Yehudayoff [40] where they called it as "mixed sources". The formal definition of interleaved sources is as follows. For a n-bit string w and a permutation $\sigma : [n] \to [n]$, we use w_σ to denote the string such that the $\sigma(i)$-th bit of w_σ is exactly the i-th bit of w. For two strings x, y, we use $x \circ y$ to denote the concatenation of x and y.

Definition 1.3. *Let* $\mathbf{X}_1$ *be an* (n, k_1)*-source,* $\mathbf{X}_2$ *be an* (n, k_2)*-source independent of* $\mathbf{X}_1$ *and* $\sigma : [2n] \to [2n]$ *be a permutation. Then* $(\mathbf{X}_1 \circ \mathbf{X}_2)_\sigma$ *is an* (n, k_1, k_2)-interleaved sources*, or an* (n, k_1)-interleaved sources *if* $k_1 = k_2$.

Such sources naturally arise in a scenario that the bits of the input source are communicated remotely to the extractor from two independent sources in an unknown (but fixed) order. Raz and Yehudayoff [40] observed that an explicit extractor for such sources yeilds a lower bound in best-partition communication complexity model.

Raz and Yehudayoff [40] constructed an extractor for $(n, (1-\beta)n)$-interleaved sources with $2^{-\Omega(n)}$ error for a small constant $\beta > 0$. Subsequently, Chattopadhyay and Zuckerman [12] constructed an extractor for $(n, (1-\gamma)n, O(\log(n)))$-interleaved sources with error $n^{-\Omega(1)}$ for a small constant $\gamma > 0$. A recent work by Chattopadhay and Li [10] gave an extractor for $(n, (2/3+\delta)n)$-interleaved sources

[2] For example, we can pick any $b \in \mathrm{Supp}(\mathbf{X})$ and take $\mathbf{X}_1 = \mathbf{X}$ and $\mathbf{X}_2 = b + \mathbf{X}$.

with error $2^{-n^{\Omega(1)}}$, where δ is an arbitrarily small constant. In summary, all prior work required at least one of the sources to have min-entropy at least $0.66n$.

Observe that interleaved sources is a special case of sumset sources, as $(\mathbf{X}_1 \circ \mathbf{X}_2)_\sigma = (\mathbf{X}_1 \circ 0^n)_\sigma + (0^n \circ \mathbf{X}_2)_\sigma$. With our extractors for sum of two sources, we obtain the first extractors for interleaved two sources with polylogarithmic entropy.

Corollary 1.4. *There exists a universal constant C such that for every $k \geq \log^C(n)$, there exists an explicit extractor $\mathrm{Ext} : \{0,1\}^n \to \{0,1\}^m$ for (n,k)-interleaved sources with error $n^{-\Omega(1)}$.*

Corollary 1.5. *For every constant $\varepsilon > 0$, there exists a constant C_ε and an explicit extractor $\mathrm{Ext} : \{0,1\}^n \to \{0,1\}$ with error ε for (n,k)-interleaved sources where*

$$k = C_\varepsilon \log(n) \log\log(n) \log\log\log^3(n).$$

We note that the above results easily extend to the setting when the two interleaved sources are of different lengths. In particular, this captures the following natural setting of "somewhere independence": suppose we have a source $\mathbf{X}$ on n bits such that for some (unknown) i, the sources $\mathbf{X}_{\leq i}$ (first i bits of $\mathbf{X}$) and $\mathbf{X}_{>i}$ (the last $n-i$ bits of $\mathbf{X}$) are independent and each have entropy at least k. As long as $k \geq \mathrm{poly}(\log n)$, we can use our sumset extractor to extract from such sources.

1.1.2 Extractors for Small-space Sources. Kamp, Rao, Vadhan and Zuckerman [26] first studied extractors for sources sampled by algorithms with limited memory. We define such small-space sources more formally as follows.

Definition 1.6. *A space-s sampling procedure $\mathcal{A}$ with n-bit output is defined as follows. For every (i,j) s.t. $i \in \mathbb{Z}, 0 \leq i < n$ and $j \in \{0,1\}^s$, let $\mathcal{D}_{i,j}$ be a distribution over $\{0,1\} \times \{0,1\}^s$. Then $\mathcal{A}$ maintains an internal state $\mathsf{state} \in \{0,1\}^s$, which is initially 0^s, and runs the following steps for time step i from 0 to $n-1$:*

(1) Sample $(x_{i+1}, \mathsf{nextstate}) \in \{0,1\} \times \{0,1\}^s$ from $\mathcal{D}_{i,\mathsf{state}}$.
(2) Output x_{i+1}, and assign $\mathsf{state} := \mathsf{nextstate}$.

Furthermore, the distribution $\mathbf{X}$ of the output $(x_1, \ldots, x_n)$ is called a space-s source.

Equivalently, a space-s source is sampled by a branching program of width 2^s (see Section 3.4 for the formal definition). In [26], they constructed an extractor for space-s source with entropy $k \geq Cn^{1-\gamma}s^\gamma$ with error $2^{-n^{\Omega(1)}}$, for a large enough constant C and a small constant $\gamma > 0$. Chattopadhyay and Li [9] then constructed an extractor with error $n^{-\Omega(1)}$ for space-s source with entropy $k \geq s^{1.1} 2^{\log^{0.51}(n)}$ based on their sumset source extractor construction. Recently, based on a new reduction to affine extractors, Chattopadhyay and Goodman [5] improved the entropy requirement to $k \geq s \cdot \mathrm{polylog}(n)$ (or $k \geq s \log^{2+o(1)}(n)$ in the constant error setting).[3]

With our new extractors for the sum of two sources, we can use the reduction in [9] to get extractors for space-s source with entropy $s\log(n) + \mathrm{polylog}(n)$, which is already an improvement over the result in [5]. In this work, we further improve the reduction and obtain the following results.

Theorem 3. *There exists a universal constant C such that for every s and every $k \geq 2s + \log^C(n)$, there exists an explicit extractor $\mathrm{Ext} : \{0,1\}^n \to \{0,1\}^m$ with error $n^{-\Omega(1)}$ and output length $m = (k-2s)^{\Omega(1)}$ for space-s sources with min-entropy k.*

Theorem 4. *For every constant $\varepsilon > 0$, there exists a constant C_ε such that there exists an explicit extractor $\mathrm{Ext} : \{0,1\}^n \to \{0,1\}$ with error ε for space-s sources with min-entropy*

$$2s + C_\varepsilon \log(n) \log\log(n) \log\log\log^3(n).$$

Interestingly, the entropy requirement of our extractors have *optimal* dependence on the space s, as Kamp, Rao, Vadhan and Zuckerman [26] showed that it is impossible to construct an extractor for space-s source with min-entropy $\leq 2s$. Moreover, the min-entropy requirement in Theorem 4 almost matches the non-constructive extractor in [26] that requires min-entropy at least $2s + O(\log(n))$.

1.2 Affine Correlation Breakers

An important building blocks in our sumset source extractor construction is an affine correlation breaker. While such an object has been constructed in previous works [6, 9, 32], in this paper we give a new construction with slightly better parameters. The main benefit of our new construction is that it is based on a *black-box reduction* from affine correlation breakers to (standard) correlation breakers, which are simpler and more well-studied. We believe this result is of independent interest.

First we define a (standard) correlation breaker. Roughly speaking, a correlation breaker takes a source $\mathbf{X}$ and a uniform seed $\mathbf{Y}$, while an adversary controls a "tampered source" $\mathbf{X}'$ correlated with $\mathbf{X}$ and a "tampered seed" $\mathbf{Y}'$ correlated with $\mathbf{Y}$. The goal of the correlation breaker is to "break the correlation" between $(\mathbf{X},\mathbf{Y})$ and $(\mathbf{X}',\mathbf{Y}')$, with the help of some "advice" α, α'. One can also consider the "multi-tampering" variant where there are many tampered sources and seeds, but in this paper we only need the single-tampering version which is defined as follows.

Definition 1.7. $\mathrm{CB} : \{0,1\}^n \times \{0,1\}^d \times \{0,1\}^a \to \{0,1\}^m$ *is a correlation breaker for entropy k with error ε (or a (k,ε)-correlation breaker for short) if for every $\mathbf{X}, \mathbf{X}' \in \{0,1\}^n$, $\mathbf{Y}, \mathbf{Y}' \in \{0,1\}^d$, $\alpha, \alpha' \in \{0,1\}^a$ such that*

- *$\mathbf{X}$ is an (n,k) source and $\mathbf{Y}$ is uniform*
- *$(\mathbf{X},\mathbf{X}')$ is independent of $(\mathbf{Y},\mathbf{Y}')$*
- *$\alpha \neq \alpha'$,*

it holds that

$$\left(\mathrm{CB}(\mathbf{X},\mathbf{Y},\alpha), \mathrm{CB}(\mathbf{X},\mathbf{Y}',\alpha')\right) \approx_\varepsilon \left(\mathbf{U}_m, \mathrm{CB}(\mathbf{X},\mathbf{Y}',\alpha')\right).$$

The first correlation breaker was constructed implicitly by Li [30] as an important building block of his independent-source extractor. Cohen [16] then formally defined and strengthened this object, and showed other interesting applications. Chattopadhyay, Goyal and Li [7] then used this object to construct the first non-malleable extractor with polylogarithmic entropy, which became a key ingredient for the two-source extractor in [13]. Correlation breakers have received a lot of attention and many new techniques were introduced to improve the construction [8, 17–20, 33, 34].

[3] Here we focus on the small-space extractors which minimize the entropy requirement. For small-space extractors with negligible error, the best known extractor roughly requires min-entropy $n^{0.51}s^{0.49}$ [5].

Affine correlation breakers were first introduced by Li in his construction of affine extractors [32], and were later used in [9] to construct sumset source extractors. An affine correlation breaker is similar to a (standard) correlation breaker, with the main difference being that it allows $\mathbf{X}$ and $\mathbf{Y}$ to have an "affine" correlation, i.e. $\mathbf{X}$ can be written as $\mathbf{A}+\mathbf{B}$ where $\mathbf{A}$ is independent of $\mathbf{Y}$ and $\mathbf{B}$ is correlated with $\mathbf{Y}$. The formal definition is as follows.

Definition 1.8. $\mathrm{AffCB}: \{0,1\}^n \times \{0,1\}^d \times \{0,1\}^a \to \{0,1\}^m$ *is a t-affine correlation breaker for entropy k with error ε (or a (t,k,ε)-affine correlation breaker for short) if for every distributions* $\mathbf{X}, \mathbf{A}, \mathbf{B} \in \{0,1\}^n$, $\mathbf{Y}, \mathbf{Y}^1, \dots, \mathbf{Y}^t \in \{0,1\}^d$ *and strings* $\alpha, \alpha^1, \dots, \alpha^t \in \{0,1\}^a$ *such that*

- $\mathbf{X} = \mathbf{A}+\mathbf{B}$
- $\mathrm{H}_\infty(\mathbf{A}) \geq k$ *and* $\mathbf{Y}$ *is uniform*
- $\mathbf{A}$ *is independent of* $(\mathbf{B}, \mathbf{Y}, \mathbf{Y}^1, \dots, \mathbf{Y}^{[t]})$
- $\forall i \in [t], \alpha \neq \alpha^i$,

it holds that

$$\left(\mathrm{AffCB}(\mathbf{X},\mathbf{Y},\alpha), \{\mathrm{AffCB}(\mathbf{X},\mathbf{Y}^i,\alpha^i)\}_{i\in[t]}\right) \approx_\gamma \left(\mathbf{U}_m, \{\mathrm{AffCB}(\mathbf{X},\mathbf{Y}^i,\alpha^i)\}_{i\in[t]}\right).$$

We say AffCB *is strong if*

$$\left(\mathrm{AffCB}(\mathbf{X},\mathbf{Y},\alpha), \mathbf{Y}, \{\mathrm{AffCB}(\mathbf{X},\mathbf{Y}^i,\alpha^i), \mathbf{Y}^i\}_{i\in[t]}\right) \approx_\gamma \left(\mathbf{U}_m, \mathbf{Y}, \{\mathrm{AffCB}(\mathbf{X},\mathbf{Y}^i,\alpha^i), \mathbf{Y}^i\}_{i\in[t]}\right).$$

The first affine correlation breaker in [32] was constructed by adapting techniques from the correlation breaker construction in [30] to the affine setting. Chattopadhyay, Goodman and Liao [6] then constructed an affine correlation breaker with better parameters based on new techniques developed in more recent works on correlation breakers [8, 16, 20, 33].

While the techniques for standard correlation breakers can be (usually) made to work for affine correlation breakers, it generally requires non-trivial modifications. Further, it is not clear whether the ideas in the standard setting can always be adapted to the affine setting. For example, the parameters of the affine correlation breaker in [6] do not match the parameters of the state-of-the-art standard correlation breaker by Li [34], because adapting the ideas in [34] to the affine setting (without loss in parameters) seems to be difficult. It is also likely that more improvements will be made in the easier setting of standard correlation breakers. Thus, we believe that a black-box reduction from affine correlation breakers to standard correlation breakers without loss in parameters will be useful. In this work, we provide such a reduction.

Theorem 5. *Let C be a large enough constant. Suppose that there exists an explicit (d_0,ε)-strong correlation breaker* $\mathrm{CB}: \{0,1\}^d \times \{0,1\}^{d_0} \times \{0,1\}^a \to \{0,1\}^{C\log^2(t+1)\log(n/\varepsilon)}$ *for some $n, t \in \mathbb{N}$. Then there exists an explicit strong t-affine correlation breaker* $\mathrm{AffCB}: \{0,1\}^n \times \{0,1\}^d \times \{0,1\}^a \to \{0,1\}^m$ *with error $O(t\varepsilon)$ for entropy*

$$k = O(td_0 + tm + t^2\log(n/\varepsilon)),$$

where

$$d = O(td_0 + m + t\log^3(t+1)\log(n/\varepsilon)).$$

As a corollary, by applying this black-box reduction on Li's correlation breaker [34], we get an affine correlation breaker with parameters slightly better than those of [6]. (See Theorem 5.5 for more details.) With the new affine correlation breaker, our extractor in Theorem 2 only requires $O(\log(n)\log\log(n)\log\log\log^3(n))$ entropy, while using the affine correlation breaker in [6] would require $O(\log(n)\log\log(n)\log\log\log^6(n))$ entropy.

In fact, if one can construct an "optimal" standard correlation breaker with entropy requirement and seed length $O(\log(n))$ (when $t = O(1), a = O(\log(n)), \varepsilon = n^{-\Omega(1)}$), which would imply a two-source extractor for entropy $O(\log(n))$, by Theorem 5 this also implies a sumset source extractor for entropy $O(\log(n))$ (which is also an affine extractor for entropy $O(\log(n))$.)

1.3 On Sumset Sources with Small Doubling

Finally we briefly discuss why a standard probabilistic method cannot be used to prove the existence of extractors for sumset sources, and show some partial results in this direction.

Suppose we want to extract from a source $\mathbf{A}+\mathbf{B}$, where $\mathbf{A}$ and $\mathbf{B}$ are independent (n,k)-sources. Without loss of generality we can assume that $\mathbf{A}$ is uniform over a set A, and $\mathbf{B}$ is uniform over another set B, such that $|A| = |B| = K$, where $K = 2^k$. A simple calculation shows that there are at most 2^{2nK} choices of sources. In a standard probabilistic argument, we would like to show that a random function[4] is an extractor for $\mathbf{A}+\mathbf{B}$ with probability at least $1-\delta$, where $\delta \ll 2^{-2nK}$, and then we could use union bound to show that a random function is an extractor for $(n,k,2)$-sources. However, this is not always true. For example, when $A = B$ is a linear subspace, then $\mathbf{A}+\mathbf{B}$ is exactly $\mathbf{A}$, which has support size K. In this case we can only guarantee that a random function is an extractor for $\mathbf{A}+\mathbf{B}$ with probability $1-2^{-\beta K}$ for some $\beta < 1$. In general, if the "entropy" of $\mathbf{A}+\mathbf{B}$ is not greater than k by too much, then the probabilistic argument above does not work.

Remark 1.9. *Note that the "bad case" is not an uncommon case that can be neglected: if we take A, B to be subsets of a linear space of dimension $k+1$, then* $|\mathrm{Supp}(\mathbf{A}+\mathbf{B})| \leq 2^{k+1}$*, which means a random function is an extractor for* $\mathbf{A}+\mathbf{B}$ *with probability at most* $1-2^{-2K}$*. However, there are roughly 2^{4K} choices of A and B, so even if we consider the bad cases separately the union bound still does not work.*

Nevertheless, we can use techniques from additive combinatorics to prove that the bad cases can be approximated with affine sources. With this result we can show that a random function is in fact a disperser[5] for sumset sources. To formally define the bad cases, first we recall the definition of sumsets from additive combinatorics (cf. [42]).

Definition 1.10. *For $A, B \subseteq \mathbb{F}_2^n$, define $A+B = \{a+b : a \in A, b \in B\}$. For A, B s.t. $|A| = |B|$ we say (A,B) has doubling constant r if $|A+B| \leq r|A|$.*

It is not hard to see that a random function is a disperser for $\mathbf{A}+\mathbf{B}$ with probability exactly $1-2^{-|A+B|+1}$. Therefore we can use union bound to show that a random function is a disperser with high

[4] A random function is sampled uniformly at random from all the possible choices of Boolean functions on n input bits.

[5] A disperser for a class of source $\mathcal{X}$ is a boolean function f which has non-constant output on the support of every $\mathbf{X} \in \mathcal{X}$.

probability for every sumset source $\mathbf{A}+\mathbf{B}$ which satisfies $|A+B| > 3n\,|A|$. When $|A+B| \le 3n\,|A|$, a celebrated result by Sanders [41] shows that $A+B$ must contain 90% of an affine subspace with dimension $\log(|A|) - O(\log^4(n))$. With the well-known fact that a random function is an extractor for affine sources with entropy $O(\log(n))$, we can conclude that a random function is a disperser for sumset source with entropy $O(\log^4(n))$.

Note that Sanders' result only guarantees that $A+B$ almost covers a large affine subspace, but this affine subspace might only be a negligible fraction of $\mathbf{A}+\mathbf{B}$. Therefore, while a random function is an extractor for affine sources, Sanders' result only implies that it is a disperser for sumset source with small doubling constant. In this paper, we prove a "distributional variant" of Sanders' result. That is, a sumset source $\mathbf{A}+\mathbf{B}$ with small doubling constant is actually statistically close to a convex combination of affine sources.

Theorem 6. *Let $\mathbf{A}, \mathbf{B}$ be uniform distribution over $A, B \subseteq \mathbb{F}_2^n$ s.t. $|A| = |B| = 2^k$ and $|A+B| \le r\,|A|$. Then $\mathbf{A}+\mathbf{B}$ is ε-close to a convex combination of affine sources with min-entropy $k - O(\varepsilon^{-2}\log(r)\log^3(r/\varepsilon))$.*

Then we get the following corollary which says that an affine extractor is also an *extractor* for sumset source with small doubling.

Corollary 1.11. *Let $\mathbf{A}, \mathbf{B}$ be uniform distribution over $A, B \subseteq \mathbb{F}_2^n$ s.t. $|A| = |B| = 2^k$ and $|A+B| \le r\,|A|$. If $\mathrm{AffExt} : \{0,1\}^n \to \{0,1\}^m$ is an extractor for affine sources with min-entropy $k - \log^4(r)$, then $\mathrm{AffExt}(\mathbf{A}+\mathbf{B})$ is $O(1)$-close to $\mathbf{U}_m$.*

We remark that while Corollary 1.11 implies that a random function is an extractor for sumset sources with small doubling, this does not mean a random function is an extractor for sumset sources in general. This is because a lower bound on $|A+B|$ is not sufficient for us to show that a random function is an extractor by probabilistic argument. (See the full version [11, Appendix B] for more discussions.)

1.4 Open Problems

In this paper we construct improved extractors for interleaved two sources and small-space sources based on our extractors for sum of two sources. Can we use our construction to get improved extractors for other classes of sources? More specifically, both of the applications require only an extractor for interleaved two sources, which is only a special case of sumset sources. Can we further exploit the generality of sumset sources?

Another natural open problem is whether we can construct sumset source extractors for $k = o(n)$ entropy with *negligible error*. Note that a special case of this problem, constructing low-error two-source extractors for $o(n)$ entropy, has been open for decades. We can also relax this problem and try to construct a low-error extractor for sum of $C > 2$ sources. This seems more achievable with current techniques, as Li [31] has already shown how to construct low-error 3-source extractors for polylogarithmic entropy. If one can solve this relaxed problem for min-entropy $k = n^{0.99}$, then it would also imply new results for affine extractors, because an affine source can be written as the sum of infinitely many independent sources.

Finally, it's also interesting to see whether a random function is an extractor for sum of two sources. In this paper we prove that sumset sources have a "structure vs randomness dichotomy": the sumset source is either close to an affine source, or has high enough entropy. In both cases a random function is a disperser. However our result does not seem strong enough to show that a random function is an extractor for sum of two sources.

Organization. In Section 2, we give an overview of the proofs for all our results. In Section 3, we introduce some necessary preliminaries and prior works. We use Section 4 to show a new reduction from small-space sources to sum of two sources which has optimal dependence on the space parameter, and prove Theorem 3 and Theorem 4. In Section 5, we show how to construct the extractors for sum of two sources in Theorem 1 and Theorem 2, assuming access to an affine correlation breaker based on Theorem 5. The formal proofs of Theorem 5 and Theorem 6 can be found in the full version [11, Section 6 and 7].

2 OVERVIEW OF PROOFS

In this section we give a high-level overview of our proofs. The overview includes some standard notations which can be found in Section 3.

2.1 Construction of Sumset Extractors

In this section we give an overview of construction of our sumset source extractors. Similar to [9], our extractor follows the two-step framework in [13]. First, we convert the sumset source into a non-oblivious bit-fixing (NOBF) source. Roughly speaking, a t-NOBF source is a string such that most of the bits are t-wise independent. (See Definition 3.19 for the formal definition.) Second, we apply known extractors for NOBF sources [13, 32, 37, 44] to get the output. In the rest of this section, we focus on the first step, which is the main contribution of this work.

2.1.1 Reduction from Two Sources. To see how our reduction works, first we recall the transformation from two independent sources to NOBF sources in [13]. Given two (n,k)-source $\mathbf{X}_1, \mathbf{X}_2$, first take a t-non-malleable extractor $\mathrm{nmExt} : \{0,1\}^n \times \{0,1\}^{d_1} \to \{0,1\}$ with error ε_1, enumerate all the seeds and output a string $\mathbf{R}_1 := \{\mathrm{nmExt}(\mathbf{X}_1, s)\}_{s\in\{0,1\}^{d_1}}$ with $D_1 = 2^{d_1}$ bits. We do not give the exact definition of non-malleable extractors here, but we need the following property proved in [13]: except for $\sqrt{\varepsilon_1}$ fraction of "bad bits", every $(t+1)$ "good bits" in $\mathbf{R}_1$ are $\sqrt{\varepsilon_1}$-close to uniform. With this property it might seem like $\mathbf{R}_1$ is close to a $(t+1)$-NOBF source, but unfortunately this is not true. While $\mathbf{R}_1$ is guaranteed to be $D_1^{t+1}\sqrt{\varepsilon_1}$-close to a NOBF source by a result in [1], this bound is trivial since $D_1 = \mathrm{poly}(1/\varepsilon_1)$. To get around this problem, [13] used the second source $\mathbf{X}_2$ to sample $D_2 \ll D_1$ bits from $\mathbf{R}_1$ and get $\mathbf{R}_2$. Now $\mathbf{R}_2$ is guaranteed to be $D_2^{t+1}\sqrt{\varepsilon_1}$-close to a NOBF source, and the error bound $D_2^{t+1}\sqrt{\varepsilon_1}$ can be very small since D_2 is decoupled from ε_1. We note that Li [31] also showed a reduction from two independent sources to NOBF sources, and the sampling step is also crucial in Li's reduction.

Chattopadhyay and Li [9] conjectured that a similar construction should work for sumset sources. However, in the setting of sumset sources, it is not clear how to perform the sampling step. For example, if one replaces both $\mathbf{X}_1$ and $\mathbf{X}_2$ in the above construction with a sumset source $\mathbf{X} = \mathbf{X}_1 + \mathbf{X}_2$, then the sampling step might not work because the randomness we use for sampling is now correlated with $\mathbf{R}_1$. Therefore, they adopted an idea in [30] which requires the given source $\mathbf{X}$ to be the sum of C independent sources, for some large enough constant C. In this paper, we show that we can actually make the sampling step work with a $(n, \mathrm{polylog}(n), 2)$-sumset source. As a result we get an extractor for sum of two independent sources.

2.1.2 Sampling with Sumset Source. As a warm up, first we assume that we are sampling from the output of a "0-non-malleable extractor", i.e. a strong seeded extractor. Let $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^{d_1} \to \{0,1\}$ be a strong seeded extractor with error ε_1. First observe that the sampling method has the following equivalent interpretation. Note that Ext and the source $\mathbf{X}_1$ together define a set of "good seeds" such that a seed s is good if $\mathrm{Ext}(\mathbf{X}_1, s)$ is $\sqrt{\varepsilon_1}$-close to uniform. Since Ext is a strong seeded extractor, $(1-\sqrt{\varepsilon_1})$ of the seeds should be good. In the sampling step we apply a sampler Samp on $\mathbf{X}_2$ to get some samples of seeds $\{\mathrm{Samp}(\mathbf{X}_2, i)\}_{i \in \{0,1\}^{d_2}}$. Then we can apply the function $\mathrm{Ext}(\mathbf{X}_1, \cdot)$ on these sampled seeds to get the output $\mathbf{R}_2 = \{\mathrm{Ext}(\mathbf{X}, \mathrm{Samp}(\mathbf{X}, i))\}_{i \in \{0,1\}^{d_2}}$ which is $2^{d_2}\sqrt{\varepsilon_1}$-close to a 1-NOBF source.

Now we move to the setting of sumset sources and replace both $\mathbf{X}_1, \mathbf{X}_2$ in the above steps with $\mathbf{X} = \mathbf{X}_1 + \mathbf{X}_2$. Our goal is to show that we can still view this reduction as if we were sampling good seeds with $\mathbf{X}_2$ and using these seeds to extract from $\mathbf{X}_1$. Consider the i-th output bit, $\mathrm{Ext}(\mathbf{X}, \mathrm{Samp}(\mathbf{X}, i))$. Our main observation is, if $\mathrm{Samp}(\cdot, i)$ is a linear function, then we can assume that we compute $\mathrm{Ext}(\mathbf{X}, \mathrm{Samp}(\mathbf{X}, i))$ in the following steps:

(1) First sample $x_2 \sim \mathbf{X}_2$.
(2) Use x_2 as the randomness of Samp to sample a "seed" $s := \mathrm{Samp}(\mathbf{X}_2, i)$.
(3) Output $\mathrm{Ext}'_{x_2,i}(\mathbf{X}_1, s) := \mathrm{Ext}(\mathbf{X}_1 + x_2, s + \mathrm{Samp}(\mathbf{X}_1, i))$.

First we claim that $\mathrm{Ext}'_{x_2,i}$ is also a strong seeded extractor. To see why this is true, observe that if we fix $\mathrm{Samp}(\mathbf{X}_1, i) = \Delta$, then $\mathrm{Ext}'_{x_2,i}(\mathbf{X}_1, \mathbf{U}) = \mathrm{Ext}(\mathbf{X}_1 + x_2, \mathbf{U} + \Delta)$. As long as $\mathbf{X}_1$ still has enough entropy after fixing $\mathrm{Samp}(\mathbf{X}_1, i)$, Ext works properly since $\mathbf{X}_1 + x_2$ is independent of $\mathbf{U} + \Delta$, $\mathbf{X}_1 + x_2$ still has enough entropy and $\mathbf{U} + \Delta$ is also uniform. Therefore, we can use $\mathrm{Ext}'_{x_2,i}$ and $\mathbf{X}_1$ to define a set of good seeds s which make $\mathrm{Ext}'_{x_2,i}(\mathbf{X}_1, s)$ close to uniform, and most of the seeds should be good. Then we can equivalently view the sampling step as if we were sampling good seeds for $\mathrm{Ext}'_{x_2,i}$ using $\mathbf{X}_2$ as the randomness.

There are still two problems left. First, the definition of $\mathrm{Ext}'_{x_2,i}$ depends on x_2, which is the randomness we use for sampling. To solve this problem, we take Ext to be linear, and prove that $(1-\sqrt{\varepsilon_1})$ fraction of the seeds s are good in the sense that $\mathrm{Ext}'_{x_2,i}(\mathbf{X}_1, s)$ is close to uniform for *every* x_2. Second, $\mathrm{Ext}'_{x_2,i}$ depends on i, which is the index of our samples. Similarly we change the definition of good seeds so that a seed s is good if $\mathrm{Ext}'_{x_2,i}(\mathbf{X}_1, s)$ is good for every x_2 and i, and by union bound we can show that $(1 - 2^{d_2}\sqrt{\varepsilon_1})$ fraction of the seeds are good. As long as $\varepsilon_1 \ll 2^{-2d_2}$, most of the seeds should be good. Now the definition of good seeds is decoupled from the sampling step, and hence we can show that most of the sampled seeds are good.

2.1.3 Sampling with Correlation Breakers. Next we turn to the case of t-non-malleable extractors. Similar to how we changed the definition of good seeds for a strong seeded extractor, we need to generalize the definition of good seeds for a non-malleable extractor in [13] to the sumset source setting. The main difference between non-malleable extractors and strong seeded extractors is, for strong seeded extractors we want a good seed to generate a uniform bit, but for t-non-malleable extractors we want a good seed to generate a bit which is uniform even when conditioned on every t other bits. Therefore the definition of good seeds for non-malleable extractors should be with respect to every possible "$(t+1)$-local view". Formally, we say a seed s is good with respect to x_2 and a set of indices $T = \{i_1, \ldots, i_{t+1}\}$ if for every $s^1, \ldots, s^t \in \{0,1\}^{d_1}$,

$$(\mathrm{nmExt}(\mathbf{X}_1 + x_2, s + \mathrm{Samp}(\mathbf{X}_1, i_1)) \approx_{\sqrt{\varepsilon_1}} \mathbf{U}_1$$

conditioned on

$$\{\mathrm{nmExt}(\mathbf{X}_1 + x_2, s^j + \mathrm{Samp}(\mathbf{X}_1, i_{j+1}))\}_{j \in [t]}.$$

Based on the proof in [13] and the arguments in the previous section, if $\mathbf{X}_1$ has enough entropy when conditioned on $\{\mathrm{Samp}(\mathbf{X}_1, i)\}_{i \in T}$, then $1 - \sqrt{\varepsilon_1}$ of the seeds are good with respect to x_2 and T. If we can prove that most of the seeds we sample using $x_2 \sim \mathbf{X}_2$ are good with respect to x_2 and every set of indices T, then the we can conclude that the output $\mathbf{R}_2 = \{\mathrm{nmExt}(\mathbf{X}, \mathrm{Samp}(\mathbf{X}, i))\}_{i \in \{0,1\}^{d_2}}$ is $D_2^{t+1}\sqrt{\varepsilon_1}$-close to a NOBF source.

Next we need to show that most of the seeds are good with respect to *every* x_2 and T, so that the sampling step is decoupled from the definition of good seeds. To deal with the dependence on T, we take the union bound over T, and we can still guarantee that $1 - D_2^{t+1}\sqrt{\varepsilon_1}$ of the seeds are good. To deal with the dependency on x_2, it suffices to replace the non-malleable extractor with a strong affine correlation breaker. Although the correlation breaker needs an additional advice string to work, here we can simply use the indices of the samples as the advice. Our final construction would be $\{\mathrm{AffCB}(\mathbf{X}, \mathrm{Samp}(\mathbf{X}, \alpha), \alpha)\}_{\alpha \in \{0,1\}^{d_2}}$.

Finally, we note that in order to make the extractor work for almost logarithmic entropy (Theorem 2), we need to replace the sampler with a "somewhere random sampler" based on the techniques in [3], and the construction and analysis should be changed correspondingly. We present the details in Section 5.

2.2 Reduction from Small-Space Sources to Sumset Sources

In this section we give an overview of our new reduction from small-space sources to sumset sources. As in all the previous works on small-space source extractors, our reduction is based on a simple fact: conditioned on the event that the sampling procedure is in state j at time i, the small-space source $\mathbf{X}$ can be divided into two independent sources $\mathbf{X}_1 \in \{0,1\}^i, \mathbf{X}_2 \in \{0,1\}^{n-i}$, such that $\mathbf{X}_1$ contains the bits generated before time i, and $\mathbf{X}_2$ contains the bits generated after time i. Kamp, Rao, Vadhan and Zuckerman [26] proved that if we pick some equally distant time steps $i_1, \ldots, i_{\ell-1}$ and condition on the states visited at these time steps, we can divide the small-space source into ℓ independent blocks such that some of them have enough entropy. However, such a reduction does not work for entropy smaller than $\sqrt{n}$ (cf. [5]). Chattopadhyay and Li [9] observed that with a sumset source extractor we can extract from the concatenation of independent sources with *unknown and uneven length*. They then showed that with a sumset source extractor, we can "adaptively" pick which time steps to condition on and break the $\sqrt{n}$ barrier. Chattopadhyay and Goodman [5] further refined this reduction and showed how to improve the entropy requirement by reducing to a convex combination of affine sources. The reductions in [9] and [5] can be viewed as "binary searching" the correct time steps to condition on, so that the given source $\mathbf{X}$ becomes the concatenation of independent blocks $(\mathbf{X}_1, \ldots, \mathbf{X}_{O(\log(n))})$ such that some of them have enough entropy. However, even though with our extractors for sum of two sources we only need two of the blocks to have enough entropy, the "binary search-based" reduction would condition on at least $\log(n)$ time steps and waste $s\log(n)$ entropy.

A possible way to improve this reduction is by directly choosing the "correct" time step to condition on so that we only get two blocks $\mathbf{X}_1 \circ \mathbf{X}_2$ both of which have enough entropy. However this is not always possible. For example, consider a distribution which is a convex combination of $\mathbf{U}_{n/2} \circ 0^{n/2}$ and $0^{n/2} \circ \mathbf{U}_{n/2}$. This distribution is a space-1 source and has entropy $n/2$, but no matter which time step we choose to condition on, one of the two blocks would have zero entropy.

To resolve these problems, we carefully define the event to condition on as follows. For ease of explanation we view the space-s sampling procedure as a branching program of width 2^s. (Unfamiliar readers can consult Section 3.4.) First, we define a vertex $v = (i, j)$ to be a "stopping vertex" if the bits generated after visiting v has entropy *less* than some threshold. Then we condition on a random variable $\mathbf{V}$ which is the *first* stopping vertex visited by the sampling process. Note that $\mathbf{V}$ is well-defined since every state at time n is a stopping vertex. Besides, conditioning on $\mathbf{V}$ only costs roughly $s + \log(n)$ entropy since there are only $n \cdot 2^s$ possible outcomes.

Now observe that the event $\mathbf{V} = (i, j)$ means the sampling process visits (i, j) but does not visit any stopping vertex before time i. We call the bits generated before time i the "first block" and the bits generated after time i the "second block". It is not hard to see that the two blocks are still independent conditioned on $\mathbf{V} = v$. Then observe that the first block has enough entropy because the second block does not contain too much entropy (by our definition of stopping vertex). Next we show that the second block also has enough entropy. For every vertex u, let $\mathbf{X}_u$ denote the bits generated after visiting u. The main observation is, if there is an edge from a vertex u to a vertex v, then unless $u \to v$ is a "bad edge" which is taken by u with probability $< \varepsilon$, the entropy of $\mathbf{X}_v$ can only be lower than $\mathbf{X}_u$ by at most $\log(1/\varepsilon)$. If we take $\varepsilon \ll 2^{-s}n^{-1}$, then by union bound the probability that any bad edge is traversed in the sampling procedure is $\ll 1$. Since we take $\mathbf{V}$ to be the *first* vertex such that $\mathbf{X}_\mathbf{V}$ has entropy lower than some threshold, the entropy of $\mathbf{X}_\mathbf{V}$ can only be $\log(1/\varepsilon) \approx s + \log(n)$ lower than the threshold. In conclusion, if we start with a space-s source with entropy roughly $2s + 2\log(n) + 2k$, and pick the entropy threshold of the second block to be roughly $k + s + \log(n)$, we can get two blocks both having entropy at least k.

2.3 From Affine to Standard Correlation Breaker

Next we briefly discuss our black-box reduction from affine correlation breakers to standard correlation breakers. To reduce an affine correlation breaker to a standard correlation breaker, our main idea is similar to that of [6]: to adapt the construction of a correlation breaker from the independent-source setting to the affine setting, we only need to make sure that every function on $\mathbf{X}$ is linear, and every function on $\mathbf{Y}$ works properly when $\mathbf{Y}$ is a weak source. However, instead of applying this idea step-by-step on existing constructions, we observe that every correlation breaker can be converted into a "two-step" construction which is easily adaptable to the affine setting. First, we take a prefix of $\mathbf{Y}$ as the seed to extract a string $\mathbf{Z}$ from $\mathbf{X}$. Next, we apply a correlation breaker which treats $\mathbf{Y}$ as the source and $\mathbf{Z}$ as the seed. This construction only computes one function on $\mathbf{X}$, which is a seeded extractor and can be replaced with a linear one. Furthermore, the remaining step (i.e. the correlation breaker) is a function on $\mathbf{Y}$, which does not need to be linear. Finally, we note that if the underlying standard correlation breaker is strong, we can use the output as the seed to extract from $\mathbf{X}$ linearly and get a strong affine correlation breaker.

A drawback of this simple reduction is that the resulting affine correlation breaker has a worse dependence on the number of tampering t. To solve this problem, we only apply this reduction when constructing a 1-affine correlation breaker based on a 1-correlation breaker. To construct a t-affine correlation breaker, we show how to strengthen a 1-affine correlation breaker to a t-affine correlation breaker based on the "independence-merging lemma" in [6]. Roughly speaking, we observe that even in the t-tampering setting, a 1-affine correlation breaker can still guarantee that the output bit is uniform when conditioned on *every single* tampered output (note that this is not true when conditioned on multiple tampered outputs *simultaneously*.) Therefore we apply $\log(t)$ rounds of alternating extractions to "merge the independence of the output bit with itself". A more detailed discussion and the formal proof can be found in the full version of this paper [11].

2.4 Sumset Sources with Small Doubling

Finally we briefly sketch how to prove that a sumset source with small doubling is close to a convex combination of affine sources. Let $A, B \subseteq \mathbb{F}_2^n$ be sets of size $K = 2^k$ and let $\mathbf{A}, \mathbf{B}$ be uniform distributions over A, B respectively. A seminal result by Sanders [41] showed that there exists a large affine subspace V such that at least $1 - \varepsilon$ fraction of V is in $A + B$. We adapt Sanders' proof to show that for every distinguisher with output range $[0, 1]$, the sumset source $\mathbf{A} + \mathbf{B}$ is indistinguishable from a convex combination of affine sources (with large entropy). Then by an application of von Neumann's minimax theorem we can find a universal convex combination of affine sources which is statistically close to $\mathbf{A} + \mathbf{B}$.

To describe the proof in more details, we first briefly recall the outline of Sanders' proof. Consider $A', B' \subseteq \mathbb{F}_2^m$ such that $|A'|, |B'| \geq \left|\mathbb{F}_2^m\right| / r$, and let $\mathbf{A}', \mathbf{B}'$ be uniform distributions over A', B' respectively. Let $\mathbb{1}_{A'+B'}$ denote the indicator function for $A' + B'$. Based on the Croot-Sisask lemma [21] and Fourier analysis, Sanders showed that for arbitrarily small constant $\varepsilon > 0$ there exists a distribution $\mathbf{T} \subseteq \mathbb{F}_2^m$ and a linear subspace V of co-dimension

$O(\log^4(r))$ s.t.

$$\mathbb{E}\left[\mathbb{1}_{A'+B'}(\mathbf{A'}+\mathbf{B'})\right] \approx_\varepsilon \mathbb{E}\left[\mathbb{1}_{A'+B'}(\mathbf{T}+\mathbf{V})\right],$$

where $\mathbf{V}$ is the uniform distribution over V. Then Sanders' original result follows directly by taking $\mathbf{T} = t$ which maximizes $\mathbb{E}\left[\mathbb{1}_{A'+B'}(t+\mathbf{V})\right]$.

A closer inspection at Sanders' proof shows that $\mathbb{1}_{A'+B'}$ can be replaced with any function $f : \mathbb{F}_2^m \to [0,1]$. (Note that the distributions $\mathbf{T}, \mathbf{V}$ depend on the function f.) This implies that $\mathbf{A'}+\mathbf{B'}$ is indistinguishable from a convex combination of affine sources by f. With our minimax argument we can conclude that $\mathbf{A'}+\mathbf{B'}$ is statistically close to a convex combination of affine sources.

However, the result above only works for dense sets A', B'. To generalize the result to sets A, B with small doubling, a standard trick in additive combinatorics is to consider a linear Freiman homomorphism $\phi : \mathbb{F}_2^n \to \mathbb{F}_2^m$, which is a linear injective function on $\ell A + \ell B$ for some constant ℓ, and consider $A' = \phi(A), B' = \phi(B)$. By considering the function $f \circ \phi^{-1}$ we can still show that

$$\mathbb{E}\left[f(\mathbf{A}+\mathbf{B})\right] = \mathbb{E}\left[f(\phi^{-1}(\mathbf{A'}+\mathbf{B'}))\right] \approx \mathbb{E}\left[f(\phi^{-1}(\mathbf{T}+\mathbf{V}))\right].$$

However, it is not clear whether $\phi^{-1}(\mathbf{T}+\mathbf{V})$ is a also a convex combination of affine sources in $\mathbb{F}_2^n$. To solve this problem, we adapt Sanders' proof to show that there exist $\mathbf{T}, \mathbf{V}$ which satisfy

$$\mathbb{E}\left[\mathbb{1}_{A'+B'}(\mathbf{A'}+\mathbf{B'})\right] \approx_\varepsilon \mathbb{E}\left[\mathbb{1}_{A'+B'}(\mathbf{T}+\mathbf{V})\right] \tag{1}$$

and

$$\mathbb{E}\left[f(\phi^{-1}(\mathbf{A'}+\mathbf{B'}))\right] \approx_\varepsilon \mathbb{E}\left[f(\phi^{-1}(\mathbf{T}+\mathbf{V}))\right] \tag{2}$$

simultaneously. This relies on a variant of the Croot-Sisask lemma which shows that there exists a large set of "common almost period" for $\mathbb{1}_{A'+B'}$ and $f \circ \phi^{-1}$. Then (1) guarantees that with probability at least $1 - 2\varepsilon$ over $t \sim \mathbf{T}$, $\phi^{-1}(t+\mathbf{V})$ is an affine source in $\mathbb{F}_2^n$ with entropy $k - O(\log^4(r))$. Therefore $\phi^{-1}(\mathbf{T}+\mathbf{V})$ is 2ε-close to a convex combination of affine sources. Finally (2) shows that $\mathbf{A}+\mathbf{B}$ is indistinguishable from $\phi^{-1}(\mathbf{T}+\mathbf{V})$ by f, which implies our claim.

3 PRELIMINARIES

In this section we introduce some preliminaries.

3.1 Notations

Basic notations. The logarithm in this paper is always base 2. For every $n \in \mathbb{N}$, define $[n] = \{1, 2, \ldots, n\}$. In this paper, $\{0,1\}^n$ and $\mathbb{F}_2^n$ are interchangeable, and so are $\{0,1\}^n$ and $[2^n]$. We use $x \circ y$ to denote the concatenation of two strings x and y. We say a function is explicit if it is computable by a polynomial time algorithm. For $x, y \in \mathbb{R}$ we use $x \approx_\varepsilon y$ to denote $|x - y| \le \varepsilon$ and $x \not\approx_\varepsilon y$ to denote $|x - y| > \varepsilon$. For every function $f : \mathcal{X} \to \mathcal{Y}$ and set $A \subseteq \mathcal{X}$, define $f(A) = \{f(x) : x \in A\}$. For a set $A \subseteq \mathcal{X}$ we use $\mathbb{1}_A : \mathcal{X} \to \{0,1\}$ to denote the indicator function of A such that $\mathbb{1}_A(x) = 1$ if and only if $x \in A$.

Distributions and random variables. We sometimes abuse notation and treat distributions and random variables as the same. We always write a random variable/distribution in boldface font. We use $\mathrm{Supp}(\mathbf{X})$ to denote the support of a distribution. We use $\mathbf{U}_n$ to denote the uniform distribution on $\{0,1\}^n$. When $\mathbf{U}_n$ appears with other random variables in the same joint distribution, $\mathbf{U}_n$ is considered to be independent of other random variables. Sometimes we omit the subscript n of $\mathbf{U}_n$ if the length is less relevant and is clear in the context.

Throughout this paper, "entropy" means min-entropy, unless specified differently.

When there is a sequence of random variables $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_t$ in the context, for every set $S \subseteq [t]$ we use $\mathbf{X}_S$ to denote the sequence of random variables which use indices in S as subscript, i.e. $\mathbf{X}_S := \{\mathbf{X}_i\}_{i \in S}$. We also use similar notation for indices on superscript.

3.2 Statistical Distance

Definition 3.1. *Let $\mathbf{D}_1, \mathbf{D}_2$ be two distributions on the same universe Ω. The* statistical distance *between $\mathbf{D}_1$ and $\mathbf{D}_2$ is*

$$\Delta(\mathbf{D}_1; \mathbf{D}_2) := \max_{T \subseteq \Omega}\Big(\Pr[\mathbf{D}_1 \in T] - \Pr[\mathbf{D}_2 \in T]\Big) = \frac{1}{2}\sum_{s \in \Omega} |\mathbf{D}_1(s) - \mathbf{D}_2(s)|.$$

We say $\mathbf{D}_1$ is ε-close to $\mathbf{D}_2$ if $\Delta(\mathbf{D}_1; \mathbf{D}_2) \le \varepsilon$, which is also denoted by $\mathbf{D}_1 \approx_\varepsilon \mathbf{D}_2$. Specifically, when there are two joint distributions $(\mathbf{X}, \mathbf{Z})$ and $(\mathbf{Y}, \mathbf{Z})$ such that $(\mathbf{X}, \mathbf{Z}) \approx_\varepsilon (\mathbf{Y}, \mathbf{Z})$, we sometimes write $(\mathbf{X} \approx_\varepsilon \mathbf{Y}) \mid \mathbf{Z}$ for short.

We frequently use the following standard properties.

Lemma 3.2. *For every distribution $\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3$ on the same universe, the following properties hold:*

- *For any distribution $\mathbf{Z}$,*
$$\Delta((\mathbf{D}_1, \mathbf{Z}); (\mathbf{D}_2, \mathbf{Z})) = \mathop{\mathbb{E}}_{z \sim \mathbf{Z}}\left[\Delta(\mathbf{D}_1|_{\mathbf{Z}=z}; \mathbf{D}_2|_{\mathbf{Z}=z})\right].$$
- *For every function f, $\Delta(f(\mathbf{D}_1); f(\mathbf{D}_2)) \le \Delta(\mathbf{D}_1; \mathbf{D}_2)$.*
- *(Triangle inequality) $\Delta(\mathbf{D}_1; \mathbf{D}_3) \le \Delta(\mathbf{D}_1; \mathbf{D}_2) + \Delta(\mathbf{D}_2; \mathbf{D}_3)$.*

3.3 Conditional Min-entropy

Definition 3.3 ([22]). *For a joint distribution $(\mathbf{X}, \mathbf{Z})$, the* average conditional min-entropy *of $\mathbf{X}$ given $\mathbf{Z}$ is*

$$\widetilde{\mathrm{H}}_\infty(\mathbf{X} \mid \mathbf{Z}) := -\log\left(\mathop{\mathbb{E}}_{z \sim \mathbf{Z}}\left[\max_x(\Pr[\mathbf{X} = x \mid \mathbf{Z} = z])\right]\right).$$

The following lemma, usually referred to as the *chain rule*, is frequently used in this paper.

Lemma 3.4 ([22]). *Let $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ be (correlated) random variables. Then*

$$\widetilde{\mathrm{H}}_\infty(\mathbf{X} \mid (\mathbf{Y}, \mathbf{Z})) \ge \widetilde{\mathrm{H}}_\infty(\mathbf{X} \mid \mathbf{Z}) - \log(\mathrm{Supp}(\mathbf{Y})).$$

When we need to consider worst-case conditional min-entropy, we use the following lemma.

Lemma 3.5 ([22]). *Let $\mathbf{X}, \mathbf{Z}$ be (correlated) random variables. For every $\varepsilon > 0$,*

$$\Pr_{z \sim \mathbf{Z}}\left[\mathrm{H}_\infty(\mathbf{X}|_{\mathbf{Z}=z}) \ge \mathrm{H}_\infty(\mathbf{X} \mid \mathbf{Z}) - \log(1/\varepsilon)\right] \ge 1 - \varepsilon.$$

Note that the above two lemmas imply the following:

Lemma 3.6 ([36]). *Let $\mathbf{X}, \mathbf{Z}$ be (correlated) random variables. For every $\varepsilon > 0$,*

$$\Pr_{z \sim \mathbf{Z}}\left[\mathrm{H}_\infty(\mathbf{X}|_{\mathbf{Z}=z}) \ge \mathrm{H}_\infty(\mathbf{X}) - \log(\mathrm{Supp}(\mathbf{Z})) - \log(1/\varepsilon)\right] \ge 1 - \varepsilon.$$

Lemma 3.7 ([22]). *Let $\varepsilon, \delta > 0$ and $\mathbf{X}, \mathbf{Z}$ be a random variables such that $\widetilde{\mathrm{H}}_\infty(\mathbf{X} \mid \mathbf{Z}) \geq k + \log(1/\delta)$. Let $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ be a (k, ε)-seeded extractor. Then*

$$(\mathrm{Ext}(\mathbf{X}, \mathbf{U}_d) \approx_{\varepsilon+\delta} \mathbf{U}_m) \mid \mathbf{Z}.$$

3.4 Branching Programs

The following definition is equivalent to Definition 1.6 in the sense that each layer corresponds to a time step and each vertex in a layer corresponds to a state in a certain time step.

Definition 3.8. *A branching program B of width w and length n (for sampling) is a directed (multi)-graph with $(n+1)$ layers $L_0, L_1, \ldots, L_n$ and has at most w vertices in each layer. The first layer (indexed by 0) has only one vertex called the start vertex, and every vertex in L_n has no outgoing edge. For every vertex v in layer $i < n$, the set of outgoing edges from v, denoted by E_v, satisfies the following.*

- *Every edge $e \in E_v$ is connected to a vertex in L_{i+1}.*
- *Each edge $e \in E_v$ is labeled with a probability, denoted by $\Pr[e]$, so that $\sum_{e \in E_v} \Pr[e] = 1$.*
- *Each edge $e \in E_v$ is labeled with a bit $b_e \in \{0,1\}$, and if two distinct edges $e_1, e_2 \in E_v$ are connected to the same vertex $w \in L_{i+1}$ then $b_{e_1} \neq b_{e_2}$. (Note that this implies $|E_v| \leq 2w$.)*

The output of B is a n-bit string generated by the following process. Let v_0 be the start vertex. Repeat the following for i from 1 to n: sample an edge $e_i \in E_{v_{i-1}}$ with probability $\Pr[e_i]$, output b_{e_i} and let v_i be the vertex which is connected by e_i. We say $(v_0, e_1, v_1, \ldots, e_n, v_n)$ is the computation path *of B. We say a random variable $\mathbf{X} \in \{0,1\}^n$ is a space-s source if it is generated by a branching program of width 2^s and length n.*

We also consider the subprograms of a branching program.

Definition 3.9. *Let $B = (L_0, L_1, \ldots, L_n)$ be a branching program of width w and length n and let v be a vertex in layer i of B. Then the subprogram of B starting at v, denoted by B_v, is the induced subgraph of B which consists of $(\{v\}, L_{i+1}, \ldots, L_n)$. Note that B_v is a branching program of width w and length $n - i$ which takes v as the start vertex.*

We need the following simple fact from [26].

Lemma 3.10 ([26]). *Let $\mathbf{X}$ be a space-s source sampled by a branching program B, and let v be a vertex in layer i of B. Then conditioned on the event that the computation path of $\mathbf{X}$ passes v, $\mathbf{X}$ is the concatenation of two independent random variables $\mathbf{X}_1 \in \{0,1\}^i$, $\mathbf{X}_2 \in \{0,1\}^{n-i}$. Moreover $\mathbf{X}_2$ is exactly the source generated by the subprogram B_v.*

3.5 Seeded Extractors

Definition 3.11. $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *is a* seeded extractor for entropy k with error ε *(or* (k, ε)-seeded extractor *for short) if for every (n, k) source $\mathbf{X}$, and every $\mathbf{Y} = \mathbf{U}_d$,*

$$\mathrm{Ext}(\mathbf{X}, \mathbf{Y}) \approx_\varepsilon \mathbf{U}_m.$$

We call d the seed length *of* Ext. *We say* Ext *is linear if* $\mathrm{Ext}(\cdot, y)$ *is a linear function for every $y \in \{0,1\}^d$. We say* Ext *is strong if*

$$(\mathrm{Ext}(\mathbf{X}, \mathbf{Y}) \approx_\varepsilon \mathbf{U}_m) \mid \mathbf{Y}.$$

Lemma 3.12 ([25]). *There exists a constant $c_{3.12}$ and a constant $\beta > 0$ such that for every $\varepsilon > 2^{-\beta n}$ and every k, there exists an explicit (k, ε)-strong seeded extractor* $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *s.t. $d = c_{3.12} \log(n/\varepsilon)$ and $m = k/2$.*

We also need the following extractor from [6] which is linear but has worse parameters.

Lemma 3.13. *There exists a constant $c_{3.13}$ such that for every $t, m \in \mathbb{N}$ and $\varepsilon > 0$, there exists an explicit $(c_{3.13}(m + \log(1/\varepsilon)), \varepsilon)$-linear strong seeded extractor* $\mathrm{LExt} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *s.t. $d = O(\frac{m}{t} + \log(n/\varepsilon) + \log^2(t)\log(m/\varepsilon))$.*

Note that when $m = t\log(n/\varepsilon)$ the seed length is bounded by $O\left(\left(\log^2(t) + 1\right)\log(n/\varepsilon)\right)$.

3.6 Samplers

First we define a sampler. Note that the definition here is different from the standard definition of averaging samplers [2] in the following sense: first, we need the sampler to work even when the given randomness is only a weak source. Second, we only care about "small tests".

Definition 3.14. $\mathrm{Samp} : \{0,1\}^n \times [D] \to \{0,1\}^m$ *is an* (ε, δ)-sampler *for entropy k if for every set $T \subseteq \{0,1\}^m$ s.t. $|T| \leq \varepsilon 2^m$ and every (n, k)-source $\mathbf{X}$,*

$$\Pr_{x \sim \mathbf{X}}\left[\Pr_{y \sim [D]}[\mathrm{Samp}(x, y) \in T] > 2\varepsilon\right] \leq \delta.$$

We say Samp *is linear if* $\mathrm{Samp}(\cdot, y)$ *is linear for every $y \in [D]$.*

Zuckerman [45] showed that one can use a seeded extractor as a sampler for weak sources.

Lemma 3.15 ([45]). *A $(k + \log(1/\delta), \varepsilon)$-seeded extractor is also an (ε, δ)-sampler for entropy k.*

The following is a relaxation of a sampler, which is called a somewhere random sampler.

Definition 3.16. $\mathrm{Samp} : \{0,1\}^n \times [D] \times [C] \to \{0,1\}^m$ *is an* (ε, δ)-somewhere random sampler *for entropy k if for every set $T \subseteq \{0,1\}^m$ s.t. $|T| \leq \varepsilon 2^m$ and every (n, k)-source $\mathbf{X}$,*

$$\Pr_{x \sim \mathbf{X}}\left[\Pr_{y \sim [D]}[\forall z \in [C]\ \mathrm{Samp}(x, y, z) \in T] > 2\varepsilon\right] \leq \delta.$$

We say Samp *is linear if* $\mathrm{Samp}(\cdot, y, z)$ *is linear for every $y \in [D], z \in [C]$.*

The following lemma is implicit in [3]. For completeness we include a proof in the full version [11, Appendix A].

Lemma 3.17 ([3]). *If there exists an explicit (ε, δ)-sampler* $\mathrm{Samp} : \{0,1\}^n \times [D_0] \to \{0,1\}^m$ *for entropy k, then for every constant $\gamma < 1$ there exists an explicit $(D^{-\gamma}, \delta)$-somewhere random sampler* $\mathrm{Samp}' : \{0,1\}^n \times [D] \times [C] \to \{0,1\}^m$ *for entropy k with $D = D_0^{O(1)}$ and $C = O\left(\frac{\log(D_0)}{\log(1/\varepsilon)}\right)$. Furthermore if* Samp *is linear then* Samp' *is also linear.*

By Lemma 3.13, Lemma 3.15 and Lemma 3.17 we can get the following explicit somewhere random smapler.

Lemma 3.18. *For every constant $\gamma < 1$, and every $\delta > 0, t < 2^{\sqrt[3]{\log(n)}}$ there exists an explicit $(D^{-\gamma}, \delta)$-linear somewhere random sampler* Samp : $\{0,1\}^n \times [D] \times [C] \to \{0,1\}^{t\log(n)}$ *for entropy $O(t\log(n)) + \log(1/\delta)$, where $D = n^{O(1)}$ and $C = O(\log^2(t))$.*

Proof. By Lemma 3.13 and Lemma 3.15, there exists an explicit (ε, δ)-linear sampler $\mathrm{Samp}' : \{0,1\}^n \times [D_0] \to \{0,1\}^{t\log(n)}$ for entropy $O(t\log(n)) + \log(1/\delta)$ where $\varepsilon = 2^{-\log(n)/\log^2(t)}$ and $D_0 = n^{O(1)}$. The claim follows by applying Lemma 3.17 on Samp'. □

3.7 Non-Oblivious Bit-Fixing Sources

Definition 3.19. *A distribution $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n)$ on $\{0,1\}^n$ is called t-wise independent if for every subset $S \subseteq [n]$ of size t we have $\mathbf{X}_S = \mathbf{U}_q$.*

Lemma 3.20 ([1]). *Let $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n)$ be a distribution on $\{0,1\}^n$. If for every $S \subseteq [n]$ s.t. $|S| \le t$,*

$$\bigoplus_{i\in S} \mathbf{X}_i \approx_\gamma \mathbf{U}_1,$$

then $\mathbf{X}$ is $2n^t\gamma$-close to a t-wise independent distribution.

Definition 3.21. *A distribution $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n)$ on $\{0,1\}^n$ is called a (q,t)-non-oblivious bit-fixing (NOBF) source if there exists a set Q s.t. $|Q| \le q$ and $\mathbf{X}_{[n]\setminus Q}$ is t-wise independent.*

In this paper we need the following extractors for NOBF sources.

Lemma 3.22 ([13, 32]). *There exists an explicit function* BFExt : $\{0,1\}^n \to \{0,1\}^m$ *for (q,t)-NOBF sources with error $n^{-\Omega(1)}$ where $m = n^{\Omega(1)}$, $q = n^{0.9}$ and $t = (m\log(n))^{C_{3.22}}$ for some constant $C_{3.22}$.*

Lemma 3.23 ([44]). *For every $\varepsilon > 0$, the majority function* Maj : $\{0,1\}^n \to \{0,1\}$ *is an extractor for (q,t)-NOBF sources with error $\varepsilon + O(n^{-0.1})$ where $q = n^{0.4}$ and $t = O(\varepsilon^{-2}\log^2(1/\varepsilon))$.*

3.8 Markov Chain

In this paper we usually consider the scenario that we have two sources $\mathbf{X}, \mathbf{Y}$ which are independent conditioned on a collection of random variables $\mathbf{Z}$. We use Markov chain as a shorthand for this.

Definition 3.24. *Let $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ be random variables. We say $\mathbf{X} \leftrightarrow \mathbf{Z} \leftrightarrow \mathbf{Y}$ is a Markov chain if $\mathbf{X}$ and $\mathbf{Y}$ are independent conditioned on any fixing of $\mathbf{Z}$.*

We frequently use the following fact.

Lemma 3.25. *If $\mathbf{X} \leftrightarrow \mathbf{Z} \leftrightarrow \mathbf{Y}$ is a Markov chain, then for every deterministic function f, let $\mathbf{W} = f(\mathbf{X}, \mathbf{Z})$. Then*

- *$(\mathbf{X}, \mathbf{W}) \leftrightarrow \mathbf{Z} \leftrightarrow \mathbf{Y}$ is a Markov chain.*
- *$\mathbf{X} \leftrightarrow (\mathbf{W}, \mathbf{Z}) \leftrightarrow \mathbf{Y}$ is a Markov chain.*

We use "$\mathbf{W}$ is a deterministic function of $\mathbf{X}$ (conditioned on $\mathbf{Z}$)" to refer to the first item, and "fix $\mathbf{W}$" to refer to the second item.

4 IMPROVED REDUCTION FOR SMALL-SPACE SOURCES

Our improved small-space extractor results are based on the following key lemma.

Lemma 4.1. *For every integer $C \ge 2$, every space-s source on n-bit with min-entropy*

$$k' \ge Ck + (C-1)\left(2s + 2\log(n/\varepsilon)\right)$$

is $(3C\varepsilon)$-close to a convex combination of (n,k,C)-sumset sources.

Note that by taking $C = 2$ in Lemma 4.1, we get that the sumset source extractor in Theorem 1 and Theorem 2 are also small-space source extractors which satisfy the parameters in Theorem 3 and Theorem 4 respectively. In the rest of this section we focus on proving Lemma 4.1. First we show how to derive Lemma 4.1 based on the following lemma.

Lemma 4.2. *Every space-s source $\mathbf{X} \in \{0,1\}^n$ with entropy at least $k = k_1 + k_2 + 2s + 2\log(n/\varepsilon)$ is 3ε-close to a convex combination of sources of the form $\mathbf{X}_1 \circ \mathbf{X}_2$ which satisfy the following properties:*

- *$\mathbf{X}_1$ is independent of $\mathbf{X}_2$*
- *$\mathrm{H}_\infty(\mathbf{X}_1) \ge k_1$, $\mathrm{H}_\infty(\mathbf{X}_2) \ge k_2$*
- *$\mathbf{X}_2$ is a space-s source*

Proof of Lemma 4.1. By induction, Lemma 4.2 implies that a space-s source with entropy $Ck + (C-1)(2s + 2\log(n/\varepsilon))$ is $3C\varepsilon$-close to a convex combination of sources of the form $\mathbf{X}_1 \circ \mathbf{X}_2 \circ \cdots \circ \mathbf{X}_C$ where $\mathbf{X}_1, \ldots, \mathbf{X}_C$ are independent, and for every $i \in [C]$, $\mathrm{H}_\infty(\mathbf{X}_i) \ge k$. Let $\ell_1, \ell_2, \ldots, \ell_C$ denote the length of $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_C$ respectively and define $p_i = \sum_{j=1}^{i-1} \ell_i$ and $s_i = \sum_{j=i+1}^{n} \ell_j$ (note that $p_1 = 0$ and $s_C = 0$). Then observe that

$$\mathbf{X}_1 \circ \cdots \circ \mathbf{X}_C = \sum_{i=1}^{C} 0^{p_i} \circ \mathbf{X}_i \circ 0^{s_i},$$

which implies that $\mathbf{X} = \mathbf{X}_1 \circ \cdots \circ \mathbf{X}_C$ is a (n,k,C)-sumset source. □

To prove Lemma 4.2, first we need the following lemma.

Lemma 4.3. *Let B be a branching program of width 2^s and length n for sampling. Let e be an edge in B connected from u to v and let $\mathbf{X}_u, \mathbf{X}_v$ be the output distributions of the subprograms B_u, B_v respectively. Then $\mathrm{H}_\infty(\mathbf{X}_v) \ge \mathrm{H}_\infty(\mathbf{X}_u) - \log(1/\Pr[e])$.*

Proof. Let $x^* = \arg\max_x \Pr[\mathbf{X}_v = x]$. Note that

$$\mathrm{H}_\infty(\mathbf{X}_v) = -\log(\Pr[\mathbf{X}_v = x^*])$$

by definition. Observe that

$$\Pr[\mathbf{X}_u = b_e \circ x^*] \ge \Pr[e] \cdot \Pr[\mathbf{X}_v = x^*].$$

Therefore,

$$\begin{aligned}\mathrm{H}_\infty(\mathbf{X}_u) &\le -\log\left(\Pr[\mathbf{X}_u = b_e \circ x^*]\right)\\ &\le -\log\left(\Pr[e]\cdot\Pr[\mathbf{X}_v = x^*]\right)\\ &= \mathrm{H}_\infty(\mathbf{X}_v) + \log(1/\Pr[e]).\end{aligned}$$

□

Next we prove Lemma 4.2.

Proof of Lemma 4.2. Let B denote the branching program that samples $\mathbf{X}$. For every v, define $\mathbf{X}_v$ to be the source generated by the subprogram B_v. Define v to be a *stopping vertex* if

$$\mathrm{H}_\infty(\mathbf{X}_v) \le k_2 + s + \log(n/\varepsilon).$$

Observe that every vertex u in the last layer is a stopping vertex since $\mathrm{H}_\infty(\mathbf{X}_u) = 0$. Therefore there is always a stopping vertex in the computation path. We define an edge e in B to be a *bad edge* if

$$\Pr[e] \leq \varepsilon/(n \cdot 2^s).$$

Now define a random variable $\mathbf{V}$ as follows:

- $\mathbf{V} = \perp$ if the computation path of $\mathbf{X}$ visits a bad edge before visiting any stopping vertex,
- otherwise, $\mathbf{V} = v$ where v is the first stopping vertex in the computation path.

Observe that $\Pr[\mathbf{V} = \perp] \leq 2\varepsilon$, since in each step of B there are at most 2^{s+1} edges starting from the current vertex, and there are n steps in total. Define

$$\mathrm{BAD} = \{v \in \mathrm{Supp}(\mathbf{V}) : \mathrm{H}_\infty(\mathbf{X}|_{\mathbf{V}=v}) \leq k - s - \log(n/\varepsilon)\}.$$

Then $\Pr[\mathbf{V} \in \mathrm{BAD}] \leq \varepsilon$ by Lemma 3.6. We claim that if $v \notin \mathrm{BAD}$ and $v \neq \perp$, then conditioned on $\mathbf{V} = v$, the source $\mathbf{X}$ can be written as $\mathbf{X}_1 \circ \mathbf{X}_2$ which satisfies the properties stated in Lemma 4.2. The claim directly implies Lemma 4.2 because $\Pr[v \in \mathrm{BAD} \vee v = \perp] \leq 3\varepsilon$ by union bound. Next we prove the claim. Let E_1 denote the event "the computation path contains v", and E_2 denote the event "the computation path does not contain any bad edge or stopping vertex before the layer of v". Observe that $\mathbf{V} = v$ is equivalent to $E_1 \wedge E_2$. Conditioned on E_1, by Lemma 3.10, $\mathbf{X}$ can be written as $\mathbf{X}_1 \circ \mathbf{X}_2$ where $\mathbf{X}_1$ is independent of $\mathbf{X}_2$ and $\mathbf{X}_2 = \mathbf{X}_v$. Now observe that E_2 only involves layers before v, so conditioned on E_1, $\mathbf{X}_2$ is independent of E_2. Therefore, conditioned on $\mathbf{V} = v$, we still have $\mathbf{X}_2 = \mathbf{X}_v$, which is a space-$s$ source, and $\mathbf{X}_1$ is still independent of $\mathbf{X}_2$. Next observe that

$$\begin{aligned}\mathrm{H}_\infty(\mathbf{X}_1) &= \mathrm{H}_\infty(\mathbf{X}|_{\mathbf{V}=v}) - \mathrm{H}_\infty(\mathbf{X}_2)\\ &\geq (k - s - \log(n/\varepsilon)) - (k_2 + s + \log(n/\varepsilon))\\ &\geq k_1.\end{aligned}$$

It remains to prove that $\mathrm{H}_\infty(\mathbf{X}_2) \geq k_2$. Assume for contradiction that $\mathrm{H}_\infty(\mathbf{X}_v) < k_2$. Let e be the edge in the computation path which connects to v, and suppose e is from u. Now consider the following two cases.

- If e is not a bad edge, then
$$\mathrm{H}_\infty(\mathbf{X}_u) \leq \mathrm{H}_\infty(\mathbf{X}_v) + \log(1/\Pr[e]) < k_2 + s + \log(n/\varepsilon),$$
which means u is also a stopping vertex. Therefore v cannot be the first stopping vertex.
- If e is a bad edge, then either there is a stopping vertex before e or $\mathbf{V} = \perp$.

In both cases $\mathbf{V} \neq v$, which is a contradiction. In conclusion we must have $\mathrm{H}_\infty(\mathbf{X}_2) \geq k_2$. □

5 EXTRACTORS FOR SUM OF TWO SOURCES

In this section we formally prove our main sumset extractor results (Theorem 1 and Theorem 2). The construction of our extractors relies on the following lemma:

Lemma 5.1 (main lemma). *For every constant $\gamma < 1$ and every $t \in \mathbb{N}$, there exists $N = n^{O(1)}$ and an explicit function* Reduce : $\{0,1\}^n \to \{0,1\}^N$ *s.t. for every $(n,k,2)$-sumset source $\mathbf{X}$, where*

$$k = O\left(t^3 \log(n) \cdot \left(\frac{\log\log(n)}{\log\log\log(n)} + \log^3(t)\right) \cdot \left(\log\log\log^4(n) + \log^4(t)\right)\right),$$

Reduce$(\mathbf{X})$ *is $N^{-\gamma}$-close to a $(N^{1-\gamma}, t)$-NOBF source.*

Before we prove Lemma 5.1, first we show how to prove Theorem 1 and Theorem 2 based on Lemma 5.1.

Proof of Theorem 1. Let Reduce : $\{0,1\}^n \to \{0,1\}^N$ be the function from Lemma 5.1 by taking $\gamma = 0.1$. Note that $N = \mathrm{poly}(n)$. Let BFExt : $\{0,1\}^N \to \{0,1\}^m$ be the NOBF-source extractor from Lemma 3.22. Let $\mathbf{X}$ be a $(n,k,2)$-source, where k is defined later. If Reduce$(\mathbf{X})$ is $N^{-\Omega(1)}$-close to a $(N^{0.9}, t)$-NOBF source where $t = (m\log(N))^{C_{3.22}}$, then

$$\mathrm{Ext}(\mathbf{X}) := \mathrm{BFExt}(\mathrm{Reduce}(\mathbf{X}))$$

is $n^{-\Omega(1)}$-close to uniform. By Lemma 5.1 it suffices to take $k = O(t^3\log^7(t)\log(n)) \leq (m\log(n))^{1+3C_{3.22}}$. □

Proof of Theorem 2. Let Reduce : $\{0,1\}^n \to \{0,1\}^N$ be the function from Lemma 5.1 by taking $\gamma = 0.6$. Note that $N = \mathrm{poly}(n)$. Let Maj : $\{0,1\}^N \to \{0,1\}$ be the NOBF-source extractor from Lemma 3.23, i.e. the majority function. Let $\mathbf{X}$ be a $(n,k,2)$-source, where k is defined later. If Reduce$(\mathbf{X})$ is $(\varepsilon/2)$-close to a $(N^{0.4}, t)$-NOBF source where $t = O(\varepsilon^{-2}\log^2(1/\varepsilon)) = O(1)$, then

$$\mathrm{Ext}(\mathbf{X}) := \mathrm{Maj}(\mathrm{Reduce}(\mathbf{X}))$$

is ε-close to uniform. By Lemma 5.1 it suffices to take

$$k = O(\log(n)\log\log(n)\log\log\log^3(n)).$$

□

Next we prove Lemma 5.1. First we recall the definition of a strong affine correlation breaker. To simplify our proof of Lemma 5.1, here we use a definition which is slightly more general than Definition 1.8.

Definition 5.2. AffCB : $\{0,1\}^n \times \{0,1\}^d \times \{0,1\}^a \to \{0,1\}^m$ *is a (t,k,γ)-affine correlation breaker if for every distribution $\mathbf{X}, \mathbf{A}, \mathbf{B} \in \{0,1\}^n$, $\mathbf{Y}, \mathbf{Y}^{[t]} \in \{0,1\}^d$, $\mathbf{Z}$ and string $\alpha, \alpha^{[t]} \in \{0,1\}^a$ s.t.*

- $\mathbf{X} = \mathbf{A} + \mathbf{B}$
- $\widetilde{\mathrm{H}}_\infty(\mathbf{A} \mid \mathbf{Z}) \geq k$
- $(\mathbf{Y}, \mathbf{Z}) = (\mathbf{U}_d, \mathbf{Z})$
- $\mathbf{A} \leftrightarrow \mathbf{Z} \leftrightarrow (\mathbf{B}, \mathbf{Y}, \mathbf{Y}^{[t]})$ *is a Markov chain*
- $\forall i \in [t]$, $\alpha \neq \alpha^i$

It holds that

$$(\mathrm{AffCB}(\mathbf{X}, \mathbf{Y}, \alpha) \approx_\gamma \mathbf{U}_m) \mid \left(\{\mathrm{AffCB}(\mathbf{X}, \mathbf{Y}^i, \alpha^i)\}_{i\in[t]}, \mathbf{Z}\right).$$

We say AffCB *is strong if*

$$(\mathrm{AffCB}(\mathbf{X}, \mathbf{Y}, \alpha) \approx_\gamma \mathbf{U}_m) \mid \left(\{\mathrm{AffCB}(\mathbf{X}, \mathbf{Y}^i, \alpha^i)\}_{i\in[t]}, \mathbf{Y}, \mathbf{Y}^{[t]}, \mathbf{Z}\right).$$

To prove Lemma 5.1, we need the following lemma, which is an analog of [13, Lemma 2.17]. Roughly speaking, we show that even if the seeds of the correlation breaker are added by some leakage from the source, most of the seeds are still good.

Lemma 5.3. *For every error parameter $\gamma > 0$ the following holds. Let*

- AffCB : $\{0,1\}^n \times \{0,1\}^d \times \{0,1\}^a \to \{0,1\}^m$ *be a (t,k,ε)-strong affine correlation breaker*
- $L : \{0,1\}^n \times \{0,1\}^a \to \{0,1\}^d$ *be any deterministic function, which we call the* leakage function

- $\alpha, \alpha^{[t]}$ *be any a-bit* advice *s.t.* $\alpha \neq \alpha^i$ *for every* $i \in [t]$
- $\mathbf{A}$ *be an* $(n, k + (t+1)\ell)$*-source*

For every $b \in \{0,1\}^n$, $y \in \{0,1\}^d$, *define*

$$\mathbf{R}_{b,y} := \mathrm{AffCB}(\mathbf{A} + b, y + L(\mathbf{A}, \alpha), \alpha)$$

and for every $i \in [t]$ *define*

$$\mathbf{R}^i_{b,y} := \mathrm{AffCB}(\mathbf{A} + b, y + L(\mathbf{A}, \alpha^i), \alpha^i).$$

Let $\mathsf{BAD}_{\alpha,\alpha^{[t]}}$ *be the set of "bad seeds", which is defined as*

$$\left\{ y \in \{0,1\}^d : \exists b, y^{[t]} \text{ s.t. } (\mathbf{R}_{b,y} \not\approx_\gamma \mathbf{U}_m) \mid \{\mathbf{R}^i_{b,y^i}\}_{i\in[t]} \right\}.$$

Then

$$\Pr_{y\sim \mathbf{U}_d}\left[y \in \mathsf{BAD}_{\alpha,\alpha^{[t]}} \right] \leq \frac{\varepsilon}{\gamma}.$$

Proof. Define deterministic functions $f^1, \ldots, f^t : \{0,1\}^d \to \{0,1\}^d$ and $g : \{0,1\}^d \to \{0,1\}^n$ s.t. for every $y \in \mathsf{BAD}_{\alpha,\alpha^{[t]}}$,

$$\left(\mathbf{R}_{g(y),y} \not\approx_\gamma \mathbf{U}_m\right) \mid \left(\{\mathbf{R}^i_{g(y),f^i(y)}\}_{i\in[t]}\right).$$

For $y \notin \mathsf{BAD}_{\alpha,\alpha^{[t]}}$ the values of $f^1(y), f^2(y), \ldots, f^t(y), g(y)$ are defined arbitrarily. Note that the existence of $f^1, \ldots, f^t, g$ is guaranteed by the definition of $\mathsf{BAD}_{\alpha,\alpha^{[t]}}$. Let $\mathbf{W} := \mathbf{U}_d$ and $\delta := \Pr\left[\mathbf{W} \in \mathsf{BAD}_{\alpha,\alpha^{[t]}}\right]$. Observe that

$$(\mathbf{R}_{g(\mathbf{W}),\mathbf{W}} \not\approx_{\gamma\delta} \mathbf{U}_m) \mid (\{\mathbf{R}^i_{g(\mathbf{W}),f^i(\mathbf{W})}\}_{i\in[t]}, \mathbf{W}).$$

Now define $\mathbf{Y} := \mathbf{W} + L(\mathbf{A}, \alpha)$, $\mathbf{Y}^i := \mathbf{W} + L(\mathbf{A}, \alpha^i)$ for every $i \in [t]$ and $\mathbf{B} := g(\mathbf{W})$. Let $\mathbf{Z} := (L(\mathbf{A}, \alpha), L(\mathbf{A}, \alpha^1), \ldots, L(\mathbf{A}, \alpha^t))$. Note that $\mathbf{Z} \in \{0,1\}^{(t+1)\ell}$ is a deterministic function of $\mathbf{A}$. With these new definitions the above equation can be rewritten as

$$(\mathrm{AffCB}(\mathbf{A} + \mathbf{B}, \mathbf{Y}, \alpha) \not\approx_{\gamma\delta} \mathbf{U}_m) \mid (\{\mathrm{AffCB}(\mathbf{A} + \mathbf{B}, \mathbf{Y}^i, \alpha^i\}_{i\in[t]}, \mathbf{W}). \tag{3}$$

Next, observe that the following conditions hold:

- $\widetilde{\mathrm{H}}_\infty(\mathbf{A} \mid \mathbf{Z}) \geq k$ (by Lemma 3.4)
- $(\mathbf{Y}, \mathbf{Z}) = (\mathbf{U}_d, \mathbf{Z})$.
- $\mathbf{A} \leftrightarrow \mathbf{Z} \leftrightarrow (\mathbf{B}, \mathbf{Y}, \mathbf{Y}^{[t]})$ is a Markov chain.

Note that the last condition holds because $\mathbf{Z}$ is a deterministic function of $\mathbf{A}$, which implies $\mathbf{A} \leftrightarrow \mathbf{Z} \leftrightarrow (\mathbf{B}, \mathbf{W})$, and $\mathbf{Y}, \mathbf{Y}^{[t]}$ are deterministic functions of $(\mathbf{Z}, \mathbf{W})$. By the definition of AffCB we have

$$(\mathrm{AffCB}(\mathbf{A} + \mathbf{B}, \mathbf{Y}, \alpha) \approx_\varepsilon \mathbf{U}_m) \mid (\{\mathrm{AffCB}(\mathbf{A} + \mathbf{B}, \mathbf{Y}^i, \alpha^i\}_{i\in[t]}, \mathbf{Y}, \mathbf{Z})$$

which implies

$$(\mathrm{AffCB}(\mathbf{A} + \mathbf{B}, \mathbf{Y}, \alpha) \approx_\varepsilon \mathbf{U}_m) \mid (\{\mathrm{AffCB}(\mathbf{A} + \mathbf{B}, \mathbf{Y}^i, \alpha^i\}_{i\in[t]}, \mathbf{W}) \tag{4}$$

since $\mathbf{W} = \mathbf{Y} - L(\mathbf{A}, \alpha)$ and $L(\mathbf{A}, \alpha)$ is a part of $\mathbf{Z}$. By (3) and (4) we get $\delta \leq \varepsilon/\gamma$. □

Next we prove the following result, which will directly imply Lemma 5.1 by plugging in proper choices of somewhere random samplers and affine correlation breakers.

Lemma 5.4. *For every* $\varepsilon, \delta > 0$ *the following holds. Let* $\mathrm{AffCB} : \{0,1\}^n \times \{0,1\}^d \times [AC] \to \{0,1\}$ *be a* $(Ct-1)$*-strong affine correlation breaker for entropy* k_1 *with error* $A^{-2t}C^{-1}\varepsilon\delta$, *and let* $\mathrm{Samp} : \{0,1\}^n \times [A] \times [C] \to \{0,1\}^d$ *be a* (ε, δ)*-somewhere random sampler for entropy* k_2. *Then for every n-bit source* $\mathbf{X} = \mathbf{X}_1 + \mathbf{X}_2$ *such that* $\mathbf{X}_1$ *is independent of* $\mathbf{X}_2$, $\mathrm{H}_\infty(\mathbf{X}_1) \geq k_1 + Ctd$ *and* $\mathrm{H}_\infty(\mathbf{X}_2) \geq k_2$, *the source*

$$\mathrm{Reduce}(\mathbf{X}) := \left\{ \bigoplus_{z\in[C]} \mathrm{AffCB}\left(\mathbf{X}, \mathrm{Samp}(\mathbf{X}, \alpha, z), (\alpha, z)\right) \right\}_{\alpha\in[A]}$$

is 3δ*-close to a convex combination of* $(2\varepsilon A, t)$*-NOBF source.*

Proof. Consider Lemma 5.3 by taking $\mathbf{X}_1$ as the source, $A^{-t}\delta$ as the error parameter and $L(x, (\alpha, z)) := \mathrm{Samp}(x, \alpha, z)$ as the leakage function. For every non-empty subset $T \subseteq [A]$ of size at most t and every $z^* \in [C]$, define a set BAD'_{T,z^*} as follows. Let α^* denote the first element in T. Let $\beta = (\alpha^*, z^*)$ and

$$\beta' = \{(\alpha, z)\}_{\alpha\in T, z\in[C]} \backslash \{\beta\}.$$

Note that β' contains at most $2^c t - 1$ advice which are all different from β. Then we define

$$\mathsf{BAD}'_{T,z^*} := \mathsf{BAD}_{\beta,\beta'},$$

where $\mathsf{BAD}_{\beta,\beta'}$ is defined as in Lemma 5.3. Observe that by definition of BAD'_{T,z^*}, for every $x_2 \in \{0,1\}^n$, if $\mathrm{Samp}(x_2, \alpha^*, z^*) \notin \mathsf{BAD}'_{T,z^*}$, then

$$\bigoplus_{\alpha\in T} \bigoplus_{z\in[C]} \mathrm{AffCB}\left(\mathbf{X}_1 + x_2, \mathrm{Samp}(\mathbf{X}_1, \alpha, z) + \mathrm{Samp}(x_2, \alpha, z), (\alpha, z)\right)$$

is $A^{-t}\delta$-close to $\mathbf{U}_1$. By the linearity of Samp, we know that for every fixing $\mathbf{X}_2 = x_2$, if $\mathrm{Samp}(x_2, \alpha^*, z^*) \notin \mathsf{BAD}'_{T,z^*}$, then

$$\left(\bigoplus_{\alpha\in T} \bigoplus_{z\in[C]} \mathrm{AffCB}\left(\mathbf{X}, \mathrm{Samp}(\mathbf{X}, \alpha, z), (\alpha, z)\right) \right) \approx_{A^{-t}\delta} \mathbf{U}_1. \tag{5}$$

By Lemma 5.3 we know that $\Pr_{y\sim\mathbf{U}_d}\left[y \in \mathsf{BAD}'_{T,z^*} \right] \leq A^{-t}C^{-1}\varepsilon$. Now define BAD' to be the union of BAD'_{T,z^*} for all possible choices of T, z^*. Since there are at most A^t choices of T and C choices of z^*, by union bound we know that $\Pr_{y\sim\mathbf{U}_d}[y \in \mathsf{BAD}'] \leq \varepsilon$. Therefore, by definition of somewhere random sampler,

$$\Pr_{x_2\sim\mathbf{X}_2}\left[\left| \{\alpha \in [A] : \forall z\ \mathrm{Samp}(x_2, \alpha, z) \in \mathsf{BAD}'\} \right| \leq 2\varepsilon A \right] \geq 1 - \delta.$$

In other words, with probability at least $1 - \delta$ over the fixing $\mathbf{X}_2 = x_2$, there exists a set $Q \subseteq [A]$ of size at most $2\varepsilon A$ which satisfies the following: for every $\alpha \in [A]\backslash Q$, there exists z_α such that $\mathrm{Samp}(x_2, \alpha, z_\alpha) \notin \mathsf{BAD}'$, which also implies $\mathrm{Samp}(x_2, \alpha, z_\alpha) \notin \mathsf{BAD}'_{T,z_\alpha}$. By Equation (5), for every $T \subseteq [A]\backslash Q$ s.t. $1 \leq |T| \leq t$,

$$\left(\bigoplus_{\alpha\in T} \bigoplus_{z\in\{0,1\}^c} \mathrm{AffCB}(\mathbf{X}, \mathrm{Samp}(\mathbf{X}, \alpha, z), (\alpha, z)) \right) \approx_{A^{-t}\delta} \mathbf{U}_1.$$

By Lemma 3.20 this implies that with probability $1 - \delta$ over the fixing of $\mathbf{X}_2$,

$$\mathrm{Reduce}(\mathbf{X}) = \left\{ \bigoplus_{z\in\{0,1\}^c} \mathrm{AffCB}(\mathbf{X}, \mathrm{Samp}(\mathbf{X}, \alpha, z), (\alpha, z)) \right\}_{\alpha\in[A]}$$

is 2δ-close to a $(2\varepsilon A, t)$-NOBF source. Therefore $\mathrm{Reduce}(\mathbf{X})$ is 3δ-close to a convex combination of $(2\varepsilon A, t)$-NOBF source. □

To get Lemma 5.1, we need the following affine correlation breaker. The formal proof of the following theorem can be found in the full version [11].

Theorem 5.5. *For every $m, a, t \in \mathbb{N}$ and $\varepsilon > 0$ there exists an explicit strong t-affine correlation breaker* $\mathrm{AffCB} : \{0,1\}^n \times \{0,1\}^d \times \{0,1\}^a \to \{0,1\}^m$ *with error ε for entropy k such that the seed length is*

$$d = O\left(t \log\left(\frac{n}{\varepsilon}\right) \cdot \left(\frac{\log(a)}{\log\log(a)} + \log^3(t)\right)\right)$$

and

$$k = O\left(tm + t\log\left(\frac{n}{\varepsilon}\right) \cdot \left(\frac{\log(a)}{\log\log(a)} + t\right)\right).$$

Proof (sketch). Apply Theorem 5 on the correlation breaker in [34]. □

Now we are ready to prove Lemma 5.1.

Proof of Lemma 5.1. Let $\mathrm{Samp} : \{0,1\}^n \times [N] \times [C] \to \{0,1\}^d$ be a $\left(\frac{N^{-\gamma}}{2}, \frac{N^{-\gamma}}{3}\right)$-somewhere random sampler from *Lemma* 3.18, where $N = n^{O(1)}$. We want to choose proper parameters d, C so that there exists a $(Ct-1)$-strong affine correlation breaker $\mathrm{AffCB} : \{0,1\}^n \times \{0,1\}^d \times [NC] \to \{0,1\}$ with error $N^{-2(t+\gamma)}C^{-1}/6$. Then Lemma 5.4 would imply Lemma 5.1. Observe that we need to guarantee

$$d \geq K_1\left(Ct^2 \log(n) \cdot \left(\frac{\log\log(n)}{\log\log\log(n)} + \log^3(Ct)\right)\right)$$

and

$$C \geq K_2 \log^2\left(\frac{d}{\log(n)}\right)$$

for some fixed constants K_1, K_2. It suffices to take

$$C = O(\log\log\log^2(n) + \log^2(t))$$

for some large enough constant factor. Then the entropy requirement of AffCB would be

$$k_1 = O\left(Ct^2 \log(n) \cdot \left(\frac{\log\log(n)}{\log\log\log(n)} + Ct\right)\right),$$

and the entropy requirement of Samp would be

$$k_2 = O(d + \log(N^\gamma)) = O(d + \log(n)).$$

To make Reduce work, the entropy of the given sumset source should be at least

$$\begin{aligned} k &= \max\{k_1 + Ctd, k_2\} \\ &= O\left(C^2 t^3 \log(n) \cdot \left(\frac{\log\log(n)}{\log\log\log(n)} + \log^3(t)\right)\right). \end{aligned}$$

Finally, observe that the running time of Reduce is N times the running time of AffCB and Samp, which is also $\mathrm{poly}(n)$. □

REFERENCES

[1] Noga Alon, Oded Goldreich, and Yishay Mansour. 2003. Almost k-wise independence versus k-wise independence. *Inf. Process. Lett.* 88, 3 (2003), 107–110. https://doi.org/10.1016/S0020-0190(03)00359-4

[2] Mihir Bellare and John Rompel. 1994. Randomness-Efficient Oblivious Sampling. In *35th Annual Symposium on Foundations of Computer Science, FOCS 1994*. IEEE Computer Society, 276–287. https://doi.org/10.1109/SFCS.1994.365687

[3] Avraham Ben-Aroya, Dean Doron, and Amnon Ta-Shma. 2019. An efficient reduction from two-source to nonmalleable extractors: achieving near-logarithmic min-entropy. *SIAM J. Comput.* (2019), STOC17–31.

[4] Eshan Chattopadhyay. 2020. Guest Column: A Recipe for Constructing Two-Source Extractors. *SIGACT News* 51, 2 (2020), 38–57. https://doi.org/10.1145/3406678.3406688

[5] Eshan Chattopadhyay and Jesse Goodman. 2021. Improved Extractors for Small-Space Sources. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021*. IEEE, 610–621. https://doi.org/10.1109/FOCS52979.2021.00066

[6] Eshan Chattopadhyay, Jesse Goodman, and Jyun-Jie Liao. 2021. Affine Extractors for Almost Logarithmic Entropy. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021*. IEEE, 622–633. https://doi.org/10.1109/FOCS52979.2021.00067

[7] Eshan Chattopadhyay, Vipul Goyal, and Xin Li. 2020. Nonmalleable Extractors and Codes, with Their Many Tampered Extensions. *SIAM J. Comput.* 49, 5 (2020), 999–1040. https://doi.org/10.1137/18M1176622

[8] Eshan Chattopadhyay and Xin Li. 2016. Explicit Non-malleable Extractors, Multi-source Extractors, and Almost Optimal Privacy Amplification Protocols. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016*. IEEE Computer Society, 158–167. https://doi.org/10.1109/FOCS.2016.25

[9] Eshan Chattopadhyay and Xin Li. 2016. Extractors for sumset sources. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016*. ACM, 299–311. https://doi.org/10.1145/2897518.2897643

[10] Eshan Chattopadhyay and Xin Li. 2020. Non-malleable Codes, Extractors and Secret Sharing for Interleaved Tampering and Composition of Tampering. In *Theory of Cryptography - 18th International Conference, TCC 2020 (Lecture Notes in Computer Science, Vol. 12552)*, Rafael Pass and Krzysztof Pietrzak (Eds.). Springer, 584–613. https://doi.org/10.1007/978-3-030-64381-2_21

[11] Eshan Chattopadhyay and Jyun-Jie Liao. 2021. Extractors for Sum of Two Sources. *CoRR* abs/2110.12652 (2021). arXiv:2110.12652 https://arxiv.org/abs/2110.12652

[12] Eshan Chattopadhyay and David Zuckerman. 2016. New Extractors for Interleaved Sources. In *31st Conference on Computational Complexity, CCC 2016 (LIPIcs, Vol. 50)*, Ran Raz (Ed.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 7:1–7:28. https://doi.org/10.4230/LIPIcs.CCC.2016.7

[13] Eshan Chattopadhyay and David Zuckerman. 2019. Explicit two-source extractors and resilient functions. *Annals of Mathematics* 189, 3 (2019), 653–705. https://www.jstor.org/stable/10.4007/annals.2019.189.3.1

[14] Benny Chor and Oded Goldreich. 1988. Unbiased Bits from Sources of Weak Randomness and Probabilistic Communication Complexity. *SIAM J. Comput.* 17, 2 (1988), 230–261. https://doi.org/10.1137/0217015

[15] Benny Chor, Oded Goldreich, Johan Håstad, Joel Friedman, Steven Rudich, and Roman Smolensky. 1985. The Bit Extraction Problem of t-Resilient Functions (Preliminary Version). In *26th Annual Symposium on Foundations of Computer Science, STOC 1985*. IEEE Computer Society, 396–407. https://doi.org/10.1109/SFCS.1985.55

[16] Gil Cohen. 2016. Local Correlation Breakers and Applications to Three-Source Extractors and Mergers. *SIAM J. Comput.* 45, 4 (2016), 1297–1338. https://doi.org/10.1137/15M1029837

[17] Gil Cohen. 2016. Making the Most of Advice: New Correlation Breakers and Their Applications. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016*. IEEE Computer Society, 188–196. https://doi.org/10.1109/FOCS.2016.28

[18] Gil Cohen. 2016. Non-Malleable Extractors - New Tools and Improved Constructions. In *31st Conference on Computational Complexity, CCC 2016 (LIPIcs, Vol. 50)*, Ran Raz (Ed.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 8:1–8:29. https://doi.org/10.4230/LIPIcs.CCC.2016.8

[19] Gil Cohen. 2017. Towards optimal two-source extractors and Ramsey graphs. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017*. ACM, 1157–1170. https://doi.org/10.1145/3055399.3055429

[20] Gil Cohen and Leonard J. Schulman. 2016. Extractors for Near Logarithmic Min-Entropy. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016*. IEEE Computer Society, 178–187. https://doi.org/10.1109/FOCS.2016.27

[21] Ernie Croot and Olof Sisask. 2010. A probabilistic technique for finding almost-periods of convolutions. *Geometric and functional analysis* 20, 6 (2010), 1367–1396.

[22] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam D. Smith. 2008. Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data. *SIAM J. Comput.* 38, 1 (2008), 97–139. https://doi.org/10.1137/060651380

[23] Zeev Dvir, Swastik Kopparty, Shubhangi Saraf, and Madhu Sudan. 2013. Extensions to the Method of Multiplicities, with Applications to Kakeya Sets and Mergers. *SIAM J. Comput.* 42, 6 (2013), 2305–2328. https://doi.org/10.1137/100783704

[24] Ariel Gabizon, Ran Raz, and Ronen Shaltiel. 2006. Deterministic Extractors for Bit-Fixing Sources by Obtaining an Independent Seed. *SIAM J. Comput.* 36, 4 (2006), 1072–1094. https://doi.org/10.1137/S0097539705447049

[25] Venkatesan Guruswami, Christopher Umans, and Salil P. Vadhan. 2009. Unbalanced expanders and randomness extractors from Parvaresh-Vardy codes. *J. ACM* 56, 4 (2009), 20:1–20:34. https://doi.org/10.1145/1538902.1538904

[26] Jesse Kamp, Anup Rao, Salil P. Vadhan, and David Zuckerman. 2011. Deterministic extractors for small-space sources. *J. Comput. Syst. Sci.* 77, 1 (2011), 191–220. https://doi.org/10.1016/j.jcss.2010.06.014

[27] Jesse Kamp and David Zuckerman. 2007. Deterministic Extractors for Bit-Fixing Sources and Exposure-Resilient Cryptography. *SIAM J. Comput.* 36, 5 (2007), 1231–1247. https://doi.org/10.1137/S0097539705446846

[28] Anatolii Alekseevich Karatsuba. 1971. On a certain arithmetic sum. In *Doklady Akademii Nauk*, Vol. 199. Russian Academy of Sciences, 770–772.

[29] Anatolii Alekseevich Karatsuba. 1991. Distribution of values of Dirichlet characters on additive sequences. In *Doklady Akademii Nauk*, Vol. 319. Russian Academy of Sciences, 543–545.

[30] Xin Li. 2013. Extractors for a Constant Number of Independent Sources with Polylogarithmic Min-Entropy. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013*. IEEE Computer Society, 100–109. https://doi.org/10.1109/FOCS.2013.19

[31] Xin Li. 2015. Non-malleable Condensers for Arbitrary Min-entropy, and Almost Optimal Protocols for Privacy Amplification. In *Theory of Cryptography - 12th Theory of Cryptography Conference, TCC 2015 (Lecture Notes in Computer Science, Vol. 9014)*. Springer, 502–531. https://doi.org/10.1007/978-3-662-46494-6_21

[32] Xin Li. 2016. Improved Two-Source Extractors, and Affine Extractors for Polylogarithmic Entropy. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016*. IEEE Computer Society, 168–177. https://doi.org/10.1109/FOCS.2016.26

[33] Xin Li. 2017. 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*. ACM, 1144–1156. https://doi.org/10.1145/3055399.3055486

[34] Xin Li. 2019. Non-Malleable Extractors and Non-Malleable Codes: Partially Optimal Constructions. In *34th Computational Complexity Conference, CCC 2019 (LIPIcs, Vol. 137)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 28:1–28:49. https://doi.org/10.4230/LIPIcs.CCC.2019.28

[35] Chi-Jen Lu, Omer Reingold, Salil P. Vadhan, and Avi Wigderson. 2003. Extractors: optimal up to constant factors. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing, STOC 2003*. ACM, 602–611. https://doi.org/10.1145/780542.780630

[36] Ueli M. Maurer and Stefan Wolf. 1997. Privacy Amplification Secure Against Active Adversaries. In *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference (Lecture Notes in Computer Science, Vol. 1294)*, Burton S. Kaliski Jr. (Ed.). Springer, 307–321. https://doi.org/10.1007/BFb0052244

[37] Raghu Meka. 2017. Explicit Resilient Functions Matching Ajtai-Linial. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017*. SIAM, 1132–1148. https://doi.org/10.1137/1.9781611974782.73

[38] Noam Nisan and David Zuckerman. 1996. Randomness is Linear in Space. *J. Comput. Syst. Sci.* 52, 1 (1996), 43–52. https://doi.org/10.1006/jcss.1996.0004

[39] Anup Rao. 2009. Extractors for Low-Weight Affine Sources. In *Proceedings of the 24th Annual IEEE Conference on Computational Complexity, CCC 2009*. IEEE Computer Society, 95–101. https://doi.org/10.1109/CCC.2009.36

[40] Ran Raz and Amir Yehudayoff. 2011. Multilinear formulas, maximal-partition discrepancy and mixed-sources extractors. *J. Comput. Syst. Sci.* 77, 1 (2011), 167–190. https://doi.org/10.1016/j.jcss.2010.06.013

[41] Tom Sanders. 2012. On the Bogolyubov-Ruzsa lemma. *Analysis & PDE* 5, 3 (2012), 627–655.

[42] Terence Tao and Van H. Vu. 2006. *Additive combinatorics*. Vol. 105. Cambridge University Press.

[43] Luca Trevisan and Salil P. Vadhan. 2000. Extracting Randomness from Samplable Distributions. In *41st Annual Symposium on Foundations of Computer Science, FOCS 2000*. IEEE Computer Society, 32–42. https://doi.org/10.1109/SFCS.2000.892063

[44] Emanuele Viola. 2014. Extractors for Circuit Sources. *SIAM J. Comput.* 43, 2 (2014), 655–672. https://doi.org/10.1137/11085983X

[45] David Zuckerman. 1997. Randomness-optimal oblivious sampling. *Random Struct. Algorithms* 11, 4 (1997), 345–367. https://doi.org/10.1002/(SICI)1098-2418(199712)11:4<345::AID-RSA4>3.0.CO;2-Z