Extractors for Low-Weight Affine Sources

Anup Rao*
Institute for Advanced Study
arao@ias.edu

November 19, 2007

Abstract

We give polynomial time computable extractors for *low-weight affince sources*. A distribution is affine if it samples a random points from some unknown low dimensional subspace of $\mathbb{F}_2^n$. A distribution is low weight affine if the corresponding linear space has a basis of low-weight vectors. Low-weight affine sources are thus a generalization of the well studied models of bit-fixing sources (which are just weight 1 affine sources).

For universal constants c, ϵ, our extractors can extract almost all the entropy from weight k^ϵ affine sources of dimension k, as long as $k > \log^c n$, with error $2^{-k^{\Omega(1)}}$. This gives new extractors for low entropy bit-fixing sources with exponentially small error, a parameter that is important for the application of these extractors to cryptography.

Our techniques involve constructing new *condensers* for *affine somewhere random sources*.

Keywords: Extractors, Affine Sources, Exposure Resilient Cryptography

*Supported by the National Science Foundation under agreement No. CCR-0324906.

1 Introduction

Fix a vector space over a finite field $\mathbb{F}^n$. Then an *affine* source with entropy k is a distribution which is uniform over some k-dimensional affine subspace of $\mathbb{F}^n$.

Definition 1.1 (Affine Source)**.** A distribution X is an entropy k affine source if there exist linearly independent vectors $\mathbf{v_0}, \dots, \mathbf{v_k} \in \mathbb{F}^n$ such that X is sampled by picking $x_1, \dots, x_k \in \mathbb{F}$ uniformly at random and computing $\mathbf{v_o} + \sum_i x_i \mathbf{v_i}$.

An extractor for entropy k affine sources is a function $\mathsf{AffExt} : \mathbb{F}^n \to \{0,1\}^m$ such that for any such source X, the distribution of $\mathsf{AffExt}(X)$ is close to the uniform distribution in statistical distance. The distance from uniform is called the *error* of the extractor. It is easy to show that a random function that outputs m bits is an extractor for affine sources with high probability, as long as $k > 2\log n$ and $m < k - O(1)$. In this paper we present work towards the goal of constructing an explicit, polynomial time computable function AffExt which is an extractor for affine sources. We focus on the case of small fields, when $\mathbb{F}$ is $GF(2)$.

Affine sources are a generalization of another class of sources called *bit-fixing*[1] sources, introduced by Chor et al. [CFG+85]. A bit-fixing source is a source giving a point in $\{0,1\}^n$ where some $n-k$ of the bits are fixed to arbitrary values, and the remaining k of the bits are distributed independently and uniformly. Thus a bit-fixing source samples a random point from an affine subspaces where every vector $\mathbf{v_1}, \dots, \mathbf{v_k}$ in the basis for the source is a weight 1 vector. A *weight* w affine source is an affine source in which every basis vector other than the shift $\mathbf{v_0}$ has at most w non-zero coordinates.

Definition 1.2 (Low-Weight Affine Source)**.** A distribution X is a weight w, entropy k affine source if there exist linearly independent vectors $\mathbf{v_0}, \dots, \mathbf{v_k} \in \mathbb{F}^n$ such that $\mathbf{v_1}, \dots, \mathbf{v_k}$ all have weight at most w, and X is sampled by picking $x_1, \dots, x_k \in \mathbb{F}$ uniformly at random to get the sample $\mathbf{v_o} + \sum_i x_i \mathbf{v_i}$.

In this paper we give new constructions of extractors for low-weight affine sources. While our techniques do not yet give extractors for general affine sources, we hope that the tools we develop here will eventually be useful in constructing such an extractor.

1.1 Applications to Cryptography

Explicit extractors for bit-fixing sources were partly motivated by applications in cryptography [CFG+85]. In normal cryptographic schemes, the security of the scheme is guaranteed as long as secret keys used in the scheme remain secret. It is natural to ask if we can guarantee security even if the adversary learns a part of the secret key. All-or-nothing transforms, introduced by Rivest [Riv97], are functions that can be used to solve this problem. These are functions that are easy to invert given the entire output, but very hard to invert given anything significantly less than the entire output. Apart from the application mentioned above, these functions have been used to give efficient block ciphers [JSY99, Bla96].

Constructions of all-or-nothing transforms appeared in [Boy99, CDH+00]. Canetti et al. [CDH+00] reduced the task of constructing these functions to the task of constructing extractors for bit-fixing sources (though there these functions are called "exposure resilient functions"). This reduction was even extended to the adaptive setting, where the adversary can decide which bit of the input to see based on the output bits that he reads. There has been a significant body of work applying extractors for bit-fixing sources to problems in cryptography [Dod00, DKM+06] and we refer the interested

[1] In this paper we only deal with *oblivious* bit-fixing sources.

reader to [Dod00] for a survey of the subject. A crucial parameter for these applications is the error of the extractor. If cryptographic schemes are to remain secure, it is important that the error of the extractors they rely on is negligible.

2 Previous Work and Our Results

Construction	Min-Entropy	Error	Output Length	Ref
Extractor for bit-fixing sources over $GF(2)$	any k	$1/\text{poly}(k)$	$\frac{\log k}{4}$	[KZ07]
Extractor for bit-fixing sources over $GF(2)$	$k > \sqrt{n}$	$2^{-\Omega(k^2/n)}$	$\Omega(k^2/n)$	[KZ07]
Extractor for bit-fixing sources over $GF(2)$	$k > \log^c(n)$ for some constant c	$1/\text{poly}(k)$	$k - o(k)$	[GRS04]
Extractor for bit-fixing sources over $GF(2)$	$k > \sqrt{n}$	$2^{-\Omega(k^2/n)}$	$k - o(k)$	[GRS04]
Extractor for affine sources over $GF(2)$	$(0.5 + \alpha)n$, for positive constant α	$2^{-\Omega(n)}$	$\Omega(n)$	[KZ]
Extractor for affine sources over a large field, $\|\mathbb{F}\| > n^{20}$	Any k	$1/\text{poly}(\|\mathbb{F}\|)$	$k - 1$ field elements	[GR05]
Disperser for affine sources over $GF(2)$	δn for any constant δ	Any constant	$\Theta(1)$	[BKS+05]
Extractor for affine sources over $GF(2)$	δn for any constant δ	$2^{-\Omega(n)}$	$\Omega(n)$	[Bou07]
Extractor for low-weight affine sources over $GF(2)$	$k > \log^c(n)$ for some constant c	$2^{-k^{\Omega(1)}}$	$k - o(k)$	This work

Table 1: Performance of extractors for affine and bit-fixing sources

Table 1 highlights some previous work for this type of problem. When the field is polynomially large in n, Gabizon and Raz [GR05] show how to extract many random bits, even when the dimension of the source is just 1.

The best known affine source extractor for constant sized fields is due to Bourgain, who gives an extractor for any linear min-entropy with exponentially small error over GF(2). Better constructions are known for the case of bit-fixing sources [KZ07, GRS04], but no extractor with negligible error for entropy $k < \sqrt{n}$ and many output bits was known even in this case.

The main result in this paper is:

Theorem 2.1. *There exist constants d, c, ϵ s.t. for every $k(n) > \log^c n$, there exists a polynomial time computable function* $\mathsf{AffExt} : \{0,1\}^n \to \{0,1\}^m$ *that is an extractor for weight $w < k^\epsilon$ affine sources over $GF(2)$ with min-entropy k, output length $m = k - o(k)$ and error 2^{-k^d}.*

Our results are an improvement to the best known extractors for bit-fixing sources, giving extractors that output almost all of the bits of entropy with negligible error, as long as k is polylogarithmically large in n, answering an open question of [KZ07].

3 Techniques

Our techniques are analogous to the techniques used to get extractors for independent sources in [Rao06].

We make progress by considering a more restricted class of affine sources, called *somewhere random affine sources*. A source is $t \times r$ affine somewhere random, if it is a distribution on $t \times r$ matrices over $GF(2)$ that is affine and one of the rows of the matrix is distributed uniformly. We will think of the number of rows of an affine somewhere random source as a measure of the quality of the source. The fewer the number of rows, the better the quality is. We will iteratively improve the quality (reduce the number of rows) of the somewhere random sources that we are working with until extracting randomness from them becomes easy. In addition to Bourgain's extractor for linear min-entropy discussed above, our construction relies on two kinds of objects:

- Our construction will use *linear strong seeded extractors* as a basic tool. These are functions $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ that have the property that for every fixed $y \in \{0,1\}^d$, $\mathsf{Ext}(\cdot, y)$ is a linear function and for any fixed source X with min-entropy k, most y's are such that $\mathsf{Ext}(X, y)$ is close to uniform. A strong seeded extractor can be viewed as a small family of deterministic functions (each function in the family indexed by a unique seed), such that for any fixed adversarially chosen source of randomness, almost all functions from the family are good extractors for that source. Linear strong seeded extractors simply give a family of *linear* functions with the same property. One example of a good linear strong seeded extractor is Trevisan's extractor [Tre01].
- Another basic tool we will use is a good *parity check matrix*. This is a linear map $P : \{0,1\}^n \to \{0,1\}^t$ with the property that $P(c) = 0$ if and only if c is 0 or has weight larger than d. We shall need to construct such P with d maximized and t minimized.

Given these two basic tools, we can describe some basic observations that go into the construction. We will then show how to put these together to get the high level view of our extractor construction.

Idea 1: There is a simple *linear condenser* for low-weight affine sources. If P is the linear function (the parity check matrix) described above, and X is any weight w affine source, $P(X)$ is another affine source with entropy roughly $d/w - 1$. To see this, observe that P is an injective function over any low-weight subspace of X of dimension w, since any such subspace only has vectors of weight a most $(d/w - 1)w < d$. Thus the dimension of $P(X)$ must be at least w.

Idea 2: We can extract random bits from *affine somewhere random sources*. This is a source sampling a boolean matrix, where one row is uniform, and every other row is dependent on the uniform row in affine ways. It turns out that we can extract from affine somewhere random sources when the source has very few rows relative to the length of each of the rows. In the extreme case, when the somewhere random source has just one row, it is a uniformly random string. When the number of rows is only a constant, we can simply use Bourgain's extractor Theorem 4.7 to get random bits. We will show how to build extractors for affine somewhere random sources even when the number of rows is polynomially related to the length of each row. We obtain the extractor by building a *condenser* for such somewhere random sources. Given an affine somewhere random source, the condenser transforms it into another affine somewhere random source with fewer rows. Here we shall be able to use the structure of the source to guarantee that parts of the source we are working with behave as if they are independent, even though they are not. Repeatedly applying the condenser reduces the number of rows until we are left with uniformly random bits.

Idea 3: The quality of affine somewhere random sources can be transferred, even when they are dependent (in affine ways). A single affine somewhere random source S with t rows can be used to convert another affine source into an affine somewhere random source with t rows, even if the two sources are dependent, as long as the number of bits that S gives is less than the entropy of the other source. We simply use the t rows of S as seeds with a linear strong seeded extractor to extract from each of the other affine sources. Although the sources are dependent, we can show that the second affine source can be written as the sum of two affine sources, one of which is independent of S. With high probability, the random row of S is a good seed to extract from this independent affine source. It turns out that the output we obtain in this way is close to a convex combination of affine somewhere random sources, each with t rows.

Our extractor is then built in the following way:

1. Use **Idea 1** to convert the input affine source into a much shorter affine source which still has entropy.

2. Use a linear strong seeded extractor to convert this short affine source into an affine somewhere random source with few ($\ll k$) rows.

3. Use **Idea 3** to transfer the quality of this affine somewhere random source back to the original affine source to get a new affine source whose rows are much longer than the length of each row.

4. Use **Idea 2** to extract from the new high quality affine somewhere random source.

The only part of the proof that uses the low-weight property of the sources we are working with is the first step.

4 Building Blocks

To save space, we defer the preliminaries of this paper to the appendix in Appendix A.

In this section we discuss the building blocks from other works that we rely on. The first concept we need is that of a *linear* seeded extractor.

Definition 4.1 (Linear Strong Seeded Extractor)**.** A function $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is a *strong seeded extractor* for min-entropy k and error ϵ if for every min-entropy k source X,

$$\Pr_{u \leftarrow_{\mathrm{R}} U_d}[|\mathsf{Ext}(X,u) - U_m| \leq \epsilon] \geq 1-\epsilon$$

where U_m is the uniform distribution on m bits. We say that the function is a *linear strong seeded extractor* if the function $\mathsf{Ext}(\cdot,u)$ is a linear function over GF(2), for every $u \in \{0,1\}^d$.

It turns out that when such extractors are used with affine sources, the output has the nice property that most of the time the error is 0. This property is not crucial to our work, but it does simply the discussion to have it.

Proposition 4.2. *Let* $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *be a strong linear seeded extractor with error* $\epsilon < 1/2$. *Let* X *be any affine source with entropy* k. *Then,*

$$\Pr_{u \leftarrow_R U_d}[|\mathsf{Ext}(X,u) - U_m| = 0] \geq 1 - \epsilon$$

Proof. Note that if X is an affine source, for every linear function $L : \{0,1\}^n \to \{0,1\}^m$, $L(X)$ is also an affine source. Thus we have that $|L(X) - U_m| = 0$ or $|L(X) - U_m| \geq 1/2$. Since for every fixed u, $\mathsf{Ext}(\cdot, u)$ is a linear function, this implies that:

$$\begin{aligned}
&\Pr_{u \leftarrow_R U_d}[|\mathsf{Ext}(X,u) - U_m| = 0] \\
&= \Pr_{u \leftarrow_R U_d}[|\mathsf{Ext}(X,u) - U_m| < 1/2] \\
&\geq \Pr_{u \leftarrow_R U_d}[|\mathsf{Ext}(X,u) - U_m| < \epsilon] \\
&\geq 1 - \epsilon
\end{aligned}$$

□

Next we list the previous constructions of seeded extractors that we will use in this paper. The following theorem was proved by Raz et al. [RRV02] building on the work of Trevisan [Tre01]:

Theorem 4.3 ([Tre01, RRV02])**.** *For every* $n, k, m \in \mathbb{N}$ *and* $\epsilon > 0$, *such that* $m \leq k \leq n$, *there is an explicit* (k, ϵ)*-strong seeded extractor* $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *with* $d = O\left(\frac{\log^2(n/\epsilon)}{\log(k/m)}\right)$.

It turns out that the extractor that the theorem gives is actually linear over GF(2). Setting the parameters appropiately, we get the following corollaries:

Corollary 4.4 ([Tre01, RRV02])**.** *For every* $n \in \mathbb{N}$, *constants* $r > 0, \gamma < 1$, *there is an explicit* (n^γ, n^{-r})*-strong linear seeded extractor* $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^{n^{\gamma'}}$ *with* $d = O(\log(n))$.

Corollary 4.5 ([Tre01, RRV02])**.** *For every* $n, k \in \mathbb{N}$, *there is an explicit* (k, ϵ)*-strong linear seeded extractor* $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^{\Omega(k)}$ *with* $d = O(\log^2(n/\epsilon))$.

If we need to get almost all of the randomness in the source out, the following corollary is available.

Corollary 4.6 ([Tre01, RRV02])**.** *For every* $n, k \in \mathbb{N}$, $\epsilon > 0$, *there is an explicit strong seeded extractor* $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^{k - O(\log^3(n/\epsilon))}$ *for min-entropy* k *and error* ϵ, *with* $d = O(\log^3(n/\epsilon))$.

4.1 Affine Source Extractors

We need the following theorem of Bourgain:

Theorem 4.7 ([Bou07])**.** *For every constant* $\delta > 0$, *there exist constants* $\gamma, \beta > 0$ *and a polynomial time computable function* $\mathsf{Bou} : \{0,1\}^n \to \{0,1\}^{\beta n}$ *s.t. for every affine source* X *of entropy* δn, $\mathsf{Bou}(X)$ *is* $2^{-\gamma n}$*-close to uniform.*

4.2 ϵ-Biased Spaces

An ϵ-Biased distribution is a distribution that is pseudorandom for linear functions.

Definition 4.8 (ϵ-Biased Distribution)**.** A distribution X over $\{0,1\}^n$ is *ϵ-biased* if for every non-zero element $v \in \{0,1\}^n$, $v \cdot X$ is ϵ-close to uniform.

Another concept we will need is the concept of ϵ-biased distributions for low weight tests.

Definition 4.9 (ϵ-Biased for Low-Weight)**.** A distribution X over $\{0,1\}^n$ is *ϵ-biased for linear tests of size w* if for every non-zero element v of $\{0,1\}^n$ whose weight is at most w, $v \cdot X$ is ϵ-close to uniform.

Explicit constructions of such distributions with very small support have been given in [NN93, AGHP92]. A construction in [AGHP92] gives a distribution that is ϵ-biased for weight w tests that can be generated using a seed of length $2 \cdot \lceil \log(1/\epsilon) + \log w + \log\log n \rceil$.

Given any such ϵ-biased distribution with small seed length, let $P : \{0,1\}^n \to \{0,1\}^t$ be the linear map whose i'th bit is the dot product of the input with the i'th element of the ϵ-biased distribution. Then we see that if $P(x) = 0$, x must have weight larger than w. In other words, P is the parity check of some code of distance larger than w.

5 Condensing Affine Somewhere Random Sources

In this section we prove the following theorem:

Theorem 5.1 (Affine Somewhere Random Extractor)**.** *There exists a polynomial time computable function* $\mathsf{Ext} : \{0,1\}^{rt} \to \{0,1\}^{r-r^{0.9}}$ *with the property that for every affine $t \times r$ somewhere random source X with $t \le r^{0.7}$,* $\mathsf{AffineExt}(X)$ *is* $2^{-r^{\Omega(1)}}$*-close to uniform.*

We shall rely on two earlier works to get our results. The first is a construction of a linear seeded extractor, mentioned in Corollary 4.6. The second is a construction of an affine source extractor for any constant entropy rate — Theorem 4.7. We will obtain our extractor by repeatedly condensing the source we are working with — starting with an affine somewhere random source, we shall iteratively reduce the number of rows in it until we are left only with random bits. We do this with the following algorithm:

Algorithm 5.2 (AffineCondense(x)).

Input: x — a $t \times r$ matrix with $t \leq r^{0.7}$.
Output: y — a $\lceil t/2 \rceil \times m$ matrix, with $m = r - r^{0.9}$.

Sub-Routines and Parameters:
Let $w = r^{0.1}$.
Let $\mathsf{Ext} : \{0,1\}^{rt} \times \{0,1\}^w \to \{0,1\}^m$ be the strong seeded extractor from Corollary 4.6, set up to extract $m = r - r^{0.9}$ bits from a min-entropy $r - 100wr^{0.7}$ source with error $\epsilon = 2^{-r^{\Omega(1)}}$.
Let $\mathsf{Bou} : \{0,1\}^{2w} \times \{0,1\}^{2w} \to \{0,1\}^d$ be the extractor from Theorem 4.7, set up to extract from entropy rate $1/2$.
Recall the definition of a *slice* of a somewhere random source — Definition A.6.

1. Let z be the $\lceil t/2 \rceil \times 2r$ matrix obtained by concatenating pairs of rows in $\mathsf{Slice}(x, w)$, i.e., the i'th row z_i is $\mathsf{Slice}(x, w)_{2i-1}, \mathsf{Slice}(x, w)_{\min\{2i,t\}}$

2. Let s be the $\lceil t/2 \rceil \times d$ matrix whose i'th row is $\mathsf{Bou}(z_i)$.

3. Let y be the $\lceil t/2 \rceil \times m$ matrix whose i'th row is $\mathsf{Ext}(x, s_i)$.

We can then show that the output of this algorithm is close to a convex combination of affine somewhere random sources:

Lemma 5.3. *For any affine $t \times r$ somewhere random source X, with $t \leq r^{0.7}$, then* AffineCondense(X) *is $2^{-r^{\Omega(1)}}$-close to a convex combination of affine somewhere random sources.*

Proof. Let $Z = \mathsf{Slice}(X, w)$ as in the algorithm. Then note that $\mathsf{Slice}(\cdot, w)$ is a linear function. Thus, by Lemma A.7, there must exist affine sources A, B with $X = A + B$, $H(B) \geq r - tw$, and $\mathsf{Slice}(B, w)$ is the all zero matrix with probability 1. In particular, this implies that $Z = \mathsf{Slice}(X, w) = \mathsf{Slice}(A, w)$ is independent of B.

Now, since X was somewhere random, there must exist an index h for which Z_h is an affine source with min-entropy rate $1/2$. Then, if β is the error of Bou, we get that:

$$|\mathsf{Bou}(Z_h) - U_d| < \beta \tag{1}$$

Since Ext is a linear seeded extractor, for any $u \in \{0,1\}^d$ we have that $\mathsf{Ext}(X, u) = \mathsf{Ext}(A + B, u) = \mathsf{Ext}(A, u) + \mathsf{Ext}(B, u)$. Note that for every fixing of Z, the output the algorithm is a linear function of the rest of the source. Thus $Y|Z = z$ is affine. All that remains to be shown is that with high probability over the choice of $z \leftarrow_{\mathrm{R}} Z$, the source $Y|Z = z$ is also somewhere random.

By Proposition 4.2, we get that

$$\begin{aligned} &\Pr_{u \leftarrow_{\mathrm{R}} U_d}[|\mathsf{Ext}(B, u) - U_m| > 0] < \epsilon \\ \Rightarrow &\Pr_{s_h \leftarrow_{\mathrm{R}} \mathsf{Bou}(Z_h)}[|\mathsf{Ext}(B, s_h) - U_m| > 0] < \epsilon + \beta \end{aligned} \tag{2}$$

Since B is independent of Z, we have that for any $z \in \mathsf{supp}(Z), u \in \{0,1\}^d$, $(\mathsf{Ext}(X, u)|Z = z) = \mathsf{Ext}(B, u) + (\mathsf{Ext}(A, u)|Z = z)$. Since A is completely determined by Z, $\mathsf{Ext}(X, u)|Z = z$ is uniform exactly when $\mathsf{Ext}(B, u)$ is uniform.

$$
\begin{aligned}
&\Pr_{z \leftarrow_R Z}[|\mathsf{Ext}(X|Z=z, \mathsf{Bou}(z_h)) - U_m| > 0] \\
&\leq \Pr_{z \leftarrow_R Z}[|\mathsf{Ext}(B, \mathsf{Bou}(z_h)) - U_m| > 0] \\
&< \epsilon + \beta && \text{by Equation 2} \\
&= 2^{-r^{\Omega(1)}}
\end{aligned}
$$

This completes the proof.

□

Given this condenser, we can use it repeatedly to get an extractor.

Algorithm 5.4 ($\mathsf{AffineSRExt}(x)$).

Input: x — a $t \times r$ matrix with $t \leq r^{0.7}$.
Output: z — an m bit string, with $m \geq r - r^{0.95}$.

1. If x has only one row, output x.
2. Else, set y to be the output of $\mathsf{AffineCondense}(x)$.
3. Set $x = y$ and go to the first step.

It's clear that the extractor succeeds. We will need to run $\mathsf{AffineCondense}$ at most $\log t$ times, which is insignificant compared to the error in each step and the reduction in the length of each of the rows. This completes the proof of Theorem 5.1.

6 Converting Low-Weight Affine Sources into Affine Somewhere Random Sources

In this section, we show how to convert any low-weight affine source, into an affine source over fewer bits that still has entropy. We simply apply the parity check matrix of a good linear error correcting code to the sample from the affine source. Suppose we are dealing with an affine source of weight w and entropy k.

Lemma 6.1. *Let $0 < \alpha < 1$ be any constant and $P : \{0,1\}^n \to \{0,1\}^t$ be the parity check function for any linear error correcting code of distance greater than wk^α. Let X be any weight w affine source with entropy k. Then $P(X)$ is an affine source with entropy at least k^α.*

Proof. First note that $P(X)$ is clearly an affine source, since it is obtained by applying a linear function to an affine source. It remains to show that $P(X)$ has the promised entropy. To see this, let $v_1, \ldots, v_k$ be a weight w basis for X. Then we see that every vector in the span of $v_1, \ldots, v_{k^\alpha}$ has weight at most wk^α. Thus, P is injective over this subspace. $P(X)$ is thus an affine source with a support of size at least 2^{k^α}, which means that $P(X)$ has entropy at least k^α. □

As our discussion in Section 4 shows, we can set $\epsilon = 1/4$ to get such a function P with output length

$$t = 2^{2\lceil \log(1/\epsilon)+\log(wk^{\alpha})+\log\log n\rceil} \leq O(w^2 k^{2\alpha} \log^2 n)$$

We can now use a linear seeded extractor to convert a low-weight affine source into an affine somewhere random source with few rows.

Algorithm 6.2 (LowConvert).
Input: $x \in \{0,1\}^n$. **Output:** z, a $\sqrt{k} \times k^{\Omega(1)}$ boolean matrix.
Sub-Routines and Parameters: Let $\alpha \in (0,1)$ be some constant that we shall set soon. Let $P : \{0,1\}^n \to \{0,1\}^{O(w^2 k^{2\alpha} \log^2 n)}$ be as in the discussion above. Let $\mathsf{Ext} : \{0,1\}^t \times \{0,1\}^{O(\log t)} \to \{0,1\}^{k^{\Omega(1)}}$ be the linear seeded extractor for min-entropy k^α promised by Corollary 4.4. We can set $w = k^{\Omega(1)}$ and α to be small enough so that the seed length is less than $\log k/2$ and the error of the extractor is less than $1/2$.
1. For every seed $i \in \{0,1\}^d$, let $z_i = \mathsf{Ext}(P(x), i)$.

Lemma 6.3. *There exists a constant $\beta < 1/2$ such that if X is any weight k^β affine source with entropy k, $\mathsf{LowConvert}(X)$ is a $\sqrt{k} \times k^\beta$ affine somewhere random source.*

Proof. By our discussion above, $P(X)$ is an affine source with entropy at least k^α. Thus the properties of Ext guarantee that one of the rows in the output is close to uniform, which implies (by Proposition 4.2) that this row is uniform. $\square$

Unfortunately, this affine somewhere random source is not good enough, since its rows are not long enough for us to apply the extractor from Theorem 5.1. Still, we can use this source to turn our original source into a somewhere random source of the right shape via the following algorithm:

Algorithm 6.4 (AffineConvert).
Input: $x \in \{0,1\}^n$. **Output:** z, a $\sqrt{k} \times m$ boolean matrix with $m = k - o(k)$.
Sub-Routines and Parameters: Let $\mathsf{LowConvert}, \beta$ be as in Lemma 6.3, set up to work with entropy k. Let $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^{k^\beta} \to \{0,1\}^m$ be the linear seeded extractor from Corollary 4.6 set up to extract $k - k^\gamma$ bits from entropy $k - k^{1/2+\beta}$ with error 2^{-k^γ} for some constant $\gamma > 0$.
1. Let z be the matrix whose i'th row is $\mathsf{Ext}(x, L(x)_i)$

We will then prove the following theorem:

Theorem 6.5. *Let X be a weight k^β affine source over $\{0,1\}^n$ with entropy k. Then* $\mathsf{AffineConvert}(X)$ *is $2^{-k^{\Omega(1)}}$-close to being a convex combination of affine somewhere random sources.*

Note that $L(X)_i$ is not independent of X, in fact it is completely determined by X! Thus it seems strange that we can prove anything about the distribution of Z. The key point is that $L(X)_i$ is a linear function of X. We can use this to show that even though these two are not independent, we can analyze them as if they are independent.

Proof. We will use Lemma A.7. By the lemma, we can write $X = A + B$ where $H(B) \geq k - k^{1/2+l}$, and B is completely independent of $L(X) = L(A)$.

Note that for any fixing of $L(X) = L(A) = s$, the output of our algorithm is an affine source.

Let h be an index such that $L(X)_h$ is uniformly random. Then we see that

$$\Pr_{s \leftarrow_{\mathrm{R}} L(A)}[|\mathsf{Ext}(B,s) - U_m| = 0] < 2^{-k^{\Omega(1)}}$$

But this implies that

$$\Pr_{s \leftarrow_{\mathrm{R}} L(X)}[|\mathsf{Ext}(X,s)|L(X) = s - U_m| = 0] < 2^{-k^{\Omega(1)}}$$

since $\mathsf{Ext}(X,s) = \mathsf{Ext}(A,s) + \mathsf{Ext}(B,s)$ and so is uniform as long as $\mathsf{Ext}(B,s)$ is uniform.

Thus for $1 - 2^{-k^{\Omega(1)}}$ fraction of s, the output is a somewhere random affine source.

□

7 The Extractor

To get the final extractor, we simply compose the algorithm from the last section with our extractor for somewhere random sources, which we discussed in Section 5.

This gives us the following theorem:

Theorem 7.1. *There exist constants $\alpha, \beta > 0$ and a polynomial time computable function* $\mathsf{AffineExt} : \{0,1\}^n \to \{0,1\}^m$ *which is an extractor for affine sources with entropy k, weight k^β, error $2^{-k^{\Omega(1)}}$ and output length $k - k^{1-\alpha}$.*

References

[AGHP92] Noga Alon, Oded Goldreich, Johan Håstad, and René Peralta. Simple construction of almost k-wise independent random variables. *Random Structures and Algorithms*, 3(3):289–304, 1992.

[BKS+05] Boaz Barak, Guy Kindler, Ronen Shaltiel, Benny Sudakov, and Avi Wigderson. Simulating independence: New constructions of condensers, Ramsey graphs, dispersers, and extractors. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 1–10, 2005.

[Bla96] Matt Blaze. High-bandwidth encryption with low-bandwidth smartcards. *Lecture Notes in Computer Science*, 1039:33–??, 1996.

[Bou07] Jean Bourgain. On the construction of affine-source extractors. *Geometric and Functional Analysis*, 1:33–57, 2007.

[Boy99] Victor Boyko. On the security properties of OAEP as an all-or-nothing transform. *Lecture Notes in Computer Science*, 1666:503–518, 1999.

[CDH+00] Ran Canetti, Yevgeniy Dodis, Shai Halevi, Eyal Kushilevitz, and Amit Sahai. Exposure-resilient functions and all-or-nothing transforms. In Bart Preneel, editor, *Advances in Cryptology — EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Computer Science*, pages 453–469. Springer-Verlag, May 2000.

[CFG+85] Benny Chor, Joel Friedman, Oded Goldreich, Johan Håstad, Steven Rudich, and Roman Smolensky. The bit extraction problem or t–resilient functions. In *Proceedings of the 26th Annual IEEE Symposium on Foundations of Computer Science*, pages 396–407, 1985.

[Dod00] Yevgeniy Dodis. *Exposure-resilient cryptography.* PhD thesis, Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, 2000.

[DKM+06] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In Serge Vaudenay, editor, *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 486–503. Springer, 2006.

[GR05] Ariel Gabizon and Ran Raz. Deterministic extractors for affine sources over large fields. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, 2005.

[GRS04] Ariel Gabizon, Ran Raz, and Ronen Shaltiel. Deterministic extractors for bit-fixing sources by obtaining an independent seed. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, 2004.

[JSY99] Markus Jakobsson, Julien P. Stern, and Moti Yung. Scramble all, encrypt small. *Lecture Notes in Computer Science*, 1636:95–111, 1999.

[KZ] Jesse Kamp and David Zuckerman. Deterministic extractors for affine sources from bent functions. *Manuscript.*

[KZ07] Jesse Kamp and David Zuckerman. Deterministic extractors for bit-fixing sources and exposure-resilient cryptography. *SIAM Journal on Computing*, 36(5):1231–1247, 2007.

[NN93] Joseph Naor and Moni Naor. Small-bias probability spaces: Efficient constructions and applications. *SIAM Journal on Computing*, 22(4):838–856, August 1993.

[Rao06] Anup Rao. Extractors for a constant number of polynomially small min-entropy independent sources. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006.

[RRV02] Ran Raz, Omer Reingold, and Salil Vadhan. Extracting all the randomness and reducing the error in trevisan's extractors. *jcss*, 65(1):97–128, 2002.

[Riv97] Ronald Rivest. All-or-nothing encryption and the package transform. *Lecture Notes in Computer Science*, 1267:210–??, 1997.

[Tre01] Luca Trevisan. Extractors and pseudorandom generators. *Journal of the ACM*, pages 860–879, 2001.

A Preliminaries

Definition A.1. Let D and F be two distributions on a set S. Their *statistical distance* is

$$|D - F| \stackrel{def}{=} \max_{T \subseteq S}(|D(T) - F(T)|) = \frac{1}{2}\sum_{s \in S}|D(s) - F(s)|$$

If $|D - F| \leq \epsilon$ we shall say that D is ϵ*-close* to F.

This measure of distance is nice because it is robust in the sense that if two distributions are close in this distance, then applying any functions to them cannot make them go further apart.

Proposition A.2. *Let D and F be any two distributions over a set S s.t. $|D - F| \leq \epsilon$. Let g be any function on S. Then $|g(D) - g(F)| \leq \epsilon$.*

Definition A.3 (Min-Entropy)**.** The *min-entropy* of a distribution X (denoted by $H_\infty(X)$) is said to be k if the heaviest point in its support has probability 2^{-k}.

Definition A.4 (Affine Source)**.** A source X is called an *affine source* if it gives uniformly random point in some affine subspace $V \subset \mathbb{F}^n$ of a vector space over a finite field $\mathbb{F}$.

Note that for an affine source X, $H_\infty(X) = H(X)$, i.e., the min-entropy and entropy are the same.

Definition A.5 (Affine Somewhere Random Source)**.** A source X is called an *affine $t \times r$ somewhere random source* if it is a an affine source giving samples which are $t \times r$ matrices with entries from a finite field $\mathbb{F}$, such that one row X_i of the source is uniformly distributed.

Sometimes our constructions will need to take a small subset of the bits of a somewhere random source, called a *slice*:

Definition A.6. Given ℓ strings of length n, $x = x_1, \ldots, x_\ell$, define $\mathsf{Slice}(x, w)$ to be the string $x' = x'_1, \ldots, x'_\ell$ such that for each i x'_i is the prefix of x_i of length w.

The following basic lemma will be key to our results about affine sources:

Lemma A.7 (Affine Conditioning)**.** *Let X be any affine source on $\{0,1\}^n$ with entropy k. Let $L : \{0,1\}^n \to \{0,1\}^m$ be any linear function. Then there exist independent affine sources A, B such that:*

- *$H(A) \leq m$.*
- *$H(B) \geq k - m$.*
- *$X = A + B$.*
- *For every $b \in \mathsf{supp}(B)$, $L(b) = 0$.*

Proof. Without loss of generality, assume the support of X is a linear subspace (if not, we can do the analysis for the corresponding linear subspace). Let B be the linear source whose support is $\{x \in \mathsf{supp}(x) : L(x) = 0\}$. Let $b_1, \ldots, b_t$ be a basis for B. Then we can complete this basis to get a basis for X. Let A be the span of the basis vectors in the completed basis that are not in B. Thus $X = B + A$.

Note that $H(A) \leq H(L(A))$ since $L(a) \neq 0$ for every $a \in \mathsf{supp}(A)$. Thus, $H(A) \leq m$. This then implies that $H(B) \geq H(X) - H(A) \geq k - m$. □