KAKEYA SETS, NEW MERGERS AND OLD EXTRACTORS

ZEEV DVIR
WEIZMANN INSTITUTE OF SCIENCE
REHOVOT, ISRAEL
ZEEV.DVIR@WEIZMANN.AC.IL
AND AVI WIGDERSON
INSTITUTE FOR ADVANCED STUDY
PRINCETON, NJ
AVI@IAS.EDU

Abstract. A merger is a probabilistic procedure which extracts the randomness out of any (arbitrarily correlated) set of random variables, as long as one of them is uniform. Our main result is an efficient, simple, optimal (to constant factors) merger, which, for k random variables on n bits each, uses a $O(\log(nk))$ seed, and whose error is $1/nk$. Our merger can be viewed as a derandomized version of the merger of Lu, Reingold, Vadhan and Wigderson (2003). Its analysis generalizes the recent resolution of the Kakeya problem in finite fields of Dvir (2008).

Following the plan set forth by Ta-Shma (1996), who defined mergers as part of this plan, our merger provides the last "missing link" to a simple and modular construction of extractors for all entropies, which is optimal to constant factors in all parameters. This complements the elegant construction of such extractors given by Guruswami, Umans and Vadhan (2007).

We also give simple extensions of our merger in two directions. First, we generalize it to handle the case where no source is uniform – in that case the merger will extract the entropy present in the most random of the given sources. Second, we observe that the merger works just as well in the computational setting, when the sources are efficiently samplable, and computational notions of entropy replace the information theoretic ones.

Key words. extractors, kakeya, derandomization.

AMS subject classifications. 68Q01, 51E99

1. Introduction.

1.1. Kakeya sets in mathematics and computer sicence. *"What is the smallest set in the plane in which one can rotate a needle around completely?"*

This natural geometric question was asked in 1917 by Japanese mathematician Soichi Kakeya. The starting point of our paper is the (more recent) finite field version of this problem and its connection to problems in mathematics and computer science.

Let $\mathbb{F}^n$ denote the n dimensional vector space over a finite field $\mathbb{F}$. A set $K \subset \mathbb{F}^n$ is called a Kakeya set if it contains a full line in every possible direction. More formally, for every (direction) $x \in \mathbb{F}^n$ there exists a point $f(x) \in \mathbb{F}^n$ such that $\{f(x) + a \cdot x | a \in \mathbb{F}\} \subset K$. The finite field Kakeya problem deals with determining the minimal size of such sets. It is usually assumed that n, the dimension, is small, and that the field is sufficiently large. A bound of $|K| > |\mathbb{F}|^{n/2}$ is easy to obtain using the simple fact that the set $\{x - y \,|\, x, y \in K\}$ fills the entire space. The best bound until recently was $\approx |\mathbb{F}|^{(4/7)\cdot n}$ [19, 17] and it uses the additive number theoretic tools developed in [2, 15]. The finite field Kakeya conjecture states that every Kakeya set K must satisfy $|K| \geq \Omega(|\mathbb{F}|^n)$, where the implied constant depends only on n.

In mathematics, the finite field Kakeya problem described above was suggested by Wolff [25] as a model problem for studying the original Euclidean Kakeya problem. A Kakeya set $K \subset \mathbb{R}^n$ is a compact subset of Euclidean space $\mathbb{R}^n$, which contains a unit line segment in every direction. Remarkably, such a set can have Lebesque measure *zero* for any dimension $n \geq 2$, as shown by Besicovitch [1]. The Euclidean Kakeya problem deals with proving lower bounds on more refined notions of 'size', such as the Minkowski or Hausdorff dimension. The Euclidean Kakeya conjecture states that

a Kakeya set $K \subset \mathbb{R}^n$ must have dimension n (which is the bound one gets for sets with positive area). Despite its recreational flavor, the Euclidean Kakeya problem is a central open problem in geometric measure theory, with deep connections to Harmonic analysis (e.g Fefferman's result on the convergence of Fourier series in high dimensions [10]) and to other important problems in analysis. Proving the Euclidean Kakeya conjecture (which is widely believed) seems notoriously difficult, and most progress on it was via combinatorial 'approximations'. For a broader perspective on the subject see the excellent surveys [25, 3, 23].

In computer science, an interest in the finite field Kakeya problem arose independently in the analysis of a construction of mergers given by [16] (in connection with extractors, which will be discussed later). Assume that we have k random variables $X_1, X_2, \ldots, X_k$, each distributed over $\mathbb{F}^n$. The X_i's are arbitrarily correlated, but we know that at least one of them is uniformly distributed (we don't know which one). We would like to have some efficient procedure that will 'merge' $X_1, \ldots, X_k$ into a single random variable Z (say, also over $\mathbb{F}^n$) in a way that will ensure that Z has high entropy. A procedure that achieves this goal is called a merger (for a formal definition of mergers and the exact notion of entropy used see Section 2). A merger can use an additional (short) random string, called a seed in order to compute the output Z (it is not hard to see that without a seed merging is impossible). The merger of [16] is computed as follows: The seed of the merger is a random vector $a = (a_1, \ldots, a_k) \in \mathbb{F}^k$ and the output is the linear combination

$$Z = a_1 \cdot X_1 + \ldots + a_k \cdot X_k.$$

That is, to merge k inputs we pick a random element in the subspace they span. What is the best lower bound on the entropy of Z? It turns out that, in order to understand this question, we must first understand the finite field Kakeya problem.

The connection between this question and the Kakeya problem can be demonstrated by the following special case. Suppose we try to 'merge' two random sources X and Y in $\mathbb{F}^n$ by outputting the linear combination

$$Z = a \cdot X + b \cdot Y,$$

where a and b are chosen uniformly (and independently) in $\mathbb{F}$. Since this computation is symmetric we can assume w.l.o.g that X is uniform. We can also assume w.l.o.g that $Y = f(X)$ for some function $f : \mathbb{F}^n \mapsto \mathbb{F}^n$ (we can condition on any additional randomness in Y). We now see that for every fixing of $b \in \mathbb{F}$, the support of the random variable Z is contained in a Kakeya set $K_b \subset \mathbb{F}^n$. We thus see that, if there existed a small Kakeya set, then we could choose the function f such that Z will have small support and so also small min-entropy. In [16] it was shown that Z has entropy *rate* (the ratio between entropy and length) at least $1/2$. This corresponds to the trivial bound of $|\mathbb{F}|^{n/2}$ on the size of Kakeya sets stated above. It was later shown in [7], using the machinery of [15], that the entropy rate of Z is at least $4/7$ (the entropy rate is the ratio between the entropy and the length, in bits, of the variable). A lower bound of ≈ 1 on the entropy rate of Z implies an optimal bound on the size of Kakeya sets.

The finite field Kakeya conjecture was proved very recently by Dvir [8].

THEOREM 1.1 ([8]). *Let $K \subset \mathbb{F}^n$ be a Kakeya set, where $\mathbb{F}$ is a finite field. Then*

$$|K| \geq C_n \cdot |\mathbb{F}|^n,$$

where C_n depends only on n. While it is not clear how this result will impact the Eucledian Kakeya problem, the proof technique of [8] is strong enough to give tight bounds on the output entropy of the [16] merger, and do much more, as we describe next.

1.2. From Kakeya sets to mergers. As was mentioned above, the technique of [8] can be used to show that the output of the [16] merger has entropy rate arbitrarily close to 1 (over sufficiently large fields). This gives a tight analysis of the performance of this merger. However, even though the [16] merger is very attractive in its simplicity, it has one major drawback – the need for k random field elements (the coefficients of the linear combination). This amount of randomness is already prohibitive if k is not constant (and indeed most of the complications in the extractor construction of [16] arise from the need to keep k constant). Certainly when k is of the same order as n the merger is completely useless, since the seed contains more entropy than the output. Reducing significantly the dependence of the seed length on k is the heart of our paper.

Our results:. Our main technical contribution is a *derandomized* version of the [16] merger which uses only a *single* field element to choose the linear combination, instead of k field elements. In other words, the k coefficients are functions of a single field element. The construction of the new merger, which we call the Curve Merger, can be described as follows (see Construction 3.1 for a more detailed description). Let $b \in \mathbb{F}$ be a random field element. The coefficients $a_1, \ldots, a_k \in \mathbb{F}$ will be given by k low degree univariate polynomials $c_1, \ldots, c_k \in \mathbb{F}[u]$ evaluated at $u = b$. The output of the merger is thus given by the linear combination

$$Z = c_1(b) \cdot X_1 + \ldots + c_k(b) \cdot X_k.$$

The polynomials $c_1, \ldots, c_k$ will be of degree $k-1$ and will satisfy the property that, for every $i \in [k]$ there exists $b_i \in \mathbb{F}$ such that when $b = b_i$ we have $Z = X_i$. In other words, the output of our merger is computed by passing a degree $k-1$ curve through the input blocks and then choosing a random point on this curve (hence the name).

The analysis of this merger generalizes the argument of [8] from families of lines to families of low degree curves. In a nutshell, the proof that the output of the merger has high entropy goes by contradiction as follows. For each $x = (x_1, \ldots, x_k)$ let C_x denote the degree $k-1$ curve passing through $x_1, \ldots, x_k$. If the output of the merger doesn't have high entropy then it must hit some relatively small set K with some noticeable probability ϵ. This means that for at least $\epsilon/2$ fraction of the inputs x, the curve C_x defined above must intersect K in at least an $\epsilon/2$ fraction of its points. Call these values of x 'good'. To derive the contradiction, we use the small size of K to design a nonzero polynomial g of relatively low degree which vanishes on K (this is the main idea taken from [8]). But for every good x, the large intersection of C_x with K guarantees that g vanishes on *all* points of the curve C_x. Now, if one of the points on the curve C_x is uniform, we can deduce that g is zero on a large ($\geq \epsilon/2$) fraction of the space. Choosing the parameters correctly we get that this implies that g is identically zero, which contradicts our assumption. The analogy with the Kakeya problem is that we have many 'directions' (the good values of x, which are many since some X_i was random and ϵ is noticable) in which low-degree curves intersect our 'Kakeya set' K in many points.

1.2.1. Extensions. We now discuss some natural generalization of our merger. In addition to the analysis above, which assumes that one of the inputs X_i is uniform

on $\mathbb{F}^n$, we also analyze the Curve Merger under the weaker assumption that some X_i has only high entropy rather then being uniform. We show that, in this case, the output of the merger has entropy rate the same as that of X_i. In other words, using our merger on *any* k distributions $X_1, \ldots, X_k$ preserves the entropy of the 'best' one.

Another simple corollary of our merger is obtained for the computational setting, in which the random variables X_i are efficiently samplable, and one of them is pseudorandom. Imagine for example that the X_i are all encryptions of the same message using different schemes, one of which is secure. Or that the X_i are supplied by different computationally bounded players, some of which are honest. In both cases we may be interested to compress these outputs, so as to preserve the pseudoentropy of the largest one. And our merger does so, with a small truly random seed (although to be honest it is not clear that minimizing the randomness can be considered an issue in such potential applications).

1.3. From mergers to extractors. We say that a random variable has min entropy $\geq k$ if the probability of it hitting any specific value is at most 2^{-k}. A (k, ϵ)-extractor is a function $E : \{0,1\}^n \times \{0,1\}^d \mapsto \{0,1\}^m$ such that for every random variable X with min entropy k, the distribution of $E(X, U_d)$ has statistical distance $\leq \epsilon$ from the uniform distribution, where U_d denotes a random variable independent of X and uniform on $\{0,1\}^d$. The input U_d is called a seed and is thought of as being much shorter (in bits) than X. Intuitively, extractors are procedures that convert 'weak' sources of randomness (distributions with some entropy) into 'strong' ones (close to uniform). Extractors (also called seeded-extractors) are extremely useful in many different areas of theoretical computer science and cryptography. Applications of extractors range from error correcting codes to expander graphs to metric embeddings to name just a few. An excellent survey of this broad field is [21].

An extractor has three interesting parameters. The first is the seed length d, which we wish to minimize. The second is the output length m, which we want to maximize (we want to have $m \approx k$). The third parameter we wish to minimize is the 'error' ϵ – the statistical distance of the output of the extractor from the uniform distribution. It can be shown, using the probabilistic method, that a random function gives an extractor which is optimal in all three parameters. This, however, is not satisfactory since in applications we need to be able to compute the extractor efficiently. An extractor which is efficiently computable is called explicit.

Since the 80's there were many papers that used a variety of techniques to construct explicit extractors (see [21] for a complete list of references). The first paper to give an explicit extractor which was optimal[1] both in seed length and in entropy output was the work of Lu, Reingold, Vadhan and Wigderson [16]. There were, however, two drawbacks to their construction. The first (and perhaps the more disturbing of the two) is that the construction was extremely complicated (compared to earlier constructions). The second problem was that the extractor was not optimal for small values of ϵ (e.g when $\epsilon = n^{-\Omega(1)}$).

Very recently, Guruswami, Umans and Vadhan [11] gave a very elegant construction of optimal extractors which uses a completely different approach than [16] (and many of its predecessors). The approach of [11] uses the strong connection between extractors and list-decodable error-correcting-codes (already present in previous works) together with the analysis of the Parvaresh-Vardy codes [18] to give a short, self

[1]For the rest of the introduction we will use the term 'optimal' to mean 'optimal up to multiplicative constant factors'.

contained, construction which achieves optimal parameters. The extractor of [11] is composed of two parts. The first part – a lossless-condenser – transforms the source X (which has min entropy k) into a 'condensed' source X', which has the same entropy as X but is now distributed over $\{0,1\}^{k'}$ with $k' < 1.01 \cdot k$. This condenser is the heart of [11] and is independently interesting. Extracting uniform bits from X' is easy and can be done in several different ways (since the entropy of X' is high relative to its length in bits).

Our results:. Using our new Curve Merger we are able to give an alternative construction of extractors which are optimal in all three parameters (for error at least polynomially large in n). This construction 'resurrects' one of the earliest approaches to the construction of extractors. This approach, due to Ta-Shma [24], is quite intuitive and can be described (informally) as follows: For every source $X \in \{0,1\}^n$ there exists a 'splitting point' $t \in [n]$ such that the partition of X into its t-bit prefix and $(n-t)$-bit suffix is a block source [2]. Extracting randomness from block sources is much easier than for general sources and was already considered in the pioneering work of Chor and Goldreich [5]. Applying an extractor for block sources on each of the n possible partitions of X (using the same seed for each partition) we obtain a distribution $Y = (Y_1, \ldots, Y_n) \in (\{0,1\}^m)^n$ such that for some $t \in [n]$, Y_t is close to uniform. The last step is to 'merge' these n blocks (using the new merger) into a single block which is has high entropy (say $0.9 \cdot m$) and then apply a simple extractor for high entropy sources to get a distribution which is close to uniform (see the comment on X' in the above discussion).

There is an intriguing superficial similarity between our merger and the [11] condenser, (as well as the Shaltiel-Umans extractor [22]), in that all use the sample to construct a low degree polynomial over a finite field, and evaluate it at random point(s). Moreover the analysis of the entropy of the output in all eventually uses the fact that low degree polynomials cannot have too many zeros. However the constructions and analysis are quite different, and (again on a superficial level), our analysis uses only fundamental properties of polynomials, while the [11] analysis relies on more complicated proeprties of finite field extensions. Our proof is also easier to grasp intuitively since it has a simple geometric interpretation derived from the Kakeya problem (a small set cannot intersect many different curves in too many points).

Finally, we feel that having two different constructions of extractors with the best known parameters more than double the chances of breaking the known bounds and obtaining even better extractors. For example, an outstanding open problem in extractor constructions is obtaining an extractor with output length $m = k$ that uses logarithmic seed. Another is achieving the optimal constant ($= 1$) in front of the seed length, which will reduce the degree of the associated bipartite graphs from polynomial to linear in the input size. We hope that our new merger will lead to progress on these problems (and others).

1.4. Subsequent work. Since the initial publication of this work, our results were improved in [6]. In particular, the bound on the size of Kakeya sets and the analysis of the curve merger were improved which resulted in a new extractor construction which can output a $1 - o(1)$ fraction of the entropy of the source while maintaining logarithmic seed length.

[2] A block source is a random variable (X_1, X_2) such that X_1 has high min-entropy and for every fixing $X_1 = x_1$ the conditional random variable $X_2|X_1 = x_1$ also has sufficiently high min-entropy.

1.5. Organization. In Section 2 we give general preliminaries and precise definitions related to extractors and mergers. In Section 3 we describe the Curve Merger and analyze its performance in Theorem 3.2. In Section 4 we show how to use the Curve Merger to construct an optimal extractor (which is given by Theorem 4.4). The extensions to the merger analysis, described in Section 1.2.1 of the introduction, are given in Section 5.

2. Preliminaries. Throughout the paper $\mathbb{F}$ will denote a finite field of q elements. For a polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ we denote by $\deg(f)$ the total degree of f. Following is a statement of the well-known Schwartz-Zippel Lemma, which bounds the number of zeros a multivariate polynomial can have.

LEMMA 2.1 (Schwartz-Zippel). *Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be a non zero polynomial with $\deg(f) \leq d$. Then*

$$|\{x \in \mathbb{F}^n \,|f(x) = 0\}| \leq d \cdot q^{n-1}.$$

2.1. Random sources and extractors. We review some of the basic definitions concerning random sources and extractors. The statistical distance between two distributions P and Q on a finite domain Ω is defined as

$$\max_{S \subseteq \Omega} |P(S) - Q(S)| .$$

We say that P is ϵ-close to Q if the statistical distance between P and Q is at most ϵ, otherwise we say that P and Q are ϵ-far. If P and Q are ϵ-close we write $P \overset{\epsilon}{\sim} Q$. We will denote by U_n a random variable distributed uniformly in $\{0,1\}^n$ and which is independent from all other variables. We say that P is a convex combination of the distributions $P_1, \ldots, P_m$ if there exist real numbers $q_1, \ldots, q_m \geq 0$ such that $\sum_{i \in [m]} q_i = 1$ for which $P = \sum_{i \in [m]} q_i \cdot P_i$. The min-entropy of a random variable X is defined as

$$\mathrm{H}_\infty(X) \triangleq \min_{x \in \mathrm{supp}(X)} \log\left(\frac{1}{\mathbf{Pr}[X = x]}\right)$$

(all logarithms are taken to the base 2 unless otherwise noted). We will call a random variable X distributed over $\{0,1\}^n$ with min-entropy k an (n,k)-source. The most 'powerful' object which is defined with relation to random sources is an extractor (also called a 'seeded' extractor).

DEFINITION 2.2 (Extractor). *A function*

$$E : \{0,1\}^n \times \{0,1\}^d \mapsto \{0,1\}^m$$

is a (k,ϵ)-extractor if for every (n,k)-source X, the distribution $E(X, U_d)$ is ϵ-close to uniform.

The next two definitions deal with a special family of sources – somewhere-random sources – and with a variant of extractors specifically tailored for this family, called mergers. Somewhere-random sources and mergers were originally defined by Ta-Shma [24]. Notice that, unlike extractors, mergers are only required to output a source with high min entropy (we can always apply an extractor for high min entropy sources to get an output which is close to uniform).

DEFINITION 2.3 (Somewhere-random source). *Let $X = (X_1, \ldots, X_k)$ be a random variable such that each X_i is distributed over $\{0,1\}^n$. We say that X is a* simple

somewhere-random source *if there exists* $i \in [k]$ *such that* X_i *is uniform. We say that* X *is a* somewhere-random source *if* X *is a convex combination of simple somewhere-random sources.*

DEFINITION 2.4 (Merger). *We say that a function*

$$M : (\{0,1\}^n)^k \times \{0,1\}^d \mapsto \{0,1\}^n$$

is an (m, ϵ)-merger *if for every somewhere-random source* $X = (X_1, \ldots, X_k)$ *such that each* X_i *is distributed over* n *bit strings, the distribution of* $M(X, U_d)$ *is* ϵ*-close to having min entropy at least* m*.*

Another family of sources, considered first by Chor and Goldreich [5], is that of block sources.

DEFINITION 2.5 (Block Source). *Let* $X = (X_1, X_2)$ *be a random variable over* $\{0,1\}^{n_1} \times \{0,1\}^{n_2}$*. We say that* X *is a* (k_1, k_2)-block source *if* X_1 *is an* (n_1, k_1)*-source and for each* $x_1 \in \{0,1\}^{n_1}$ *the conditional random variable* $X_2 | X_1 = x_1$ *is an* (n_2, k_2)*-source. A function* $E : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \times \{0,1\}^d \mapsto \{0,1\}^m$ *is a* (k_1, k_2, ϵ)-block source extractor *if for every* (k_1, k_2)*-block source* X*, the distribution* $E(X, U_d)$ *is* ϵ*-close to uniform.*

DEFINITION 2.6 (Somewhere block source). *Let* $X = (X_1, \ldots, X_k)$ *be a random variable such that each* X_i *is distributed over* $\{0,1\}^{n_{i,1}} \times \{0,1\}^{n_{i,2}}$*. We say that* X *is a* simple (k_1, k_2)-somewhere block source *if there exists* $i \in [k]$ *such that* X_i *is a* (k_1, k_2)*-block source. We say that* X *is a* somewhere (k_1, k_2)-block source *if* X *is a convex combination of simple* (k_1, k_2)*-somewhere block sources.*

We will call an extractor (or merger) explicit if they can be computed using a deterministic polynomial time Turing machine. [3]

3. The Curve Merger. In this section we describe our new merger and analyze its performance. We call this merger the 'Curve Merger' since the output of the merger is computed by passing a degree $k-1$ curve through the k input blocks $x_1, \ldots, x_k \in \mathbb{F}^r$ (r will be chosen such that $r \cdot \log(q) = n$) and then using the 'seed' (which is an element of $\mathbb{F}$) to choose a point uniformly on this curve.

CONSTRUCTION 3.1 (Curve Merger). *Let* $\gamma_1, \ldots, \gamma_k \in \mathbb{F}$ *be* k *distinct field elements. We define* k *univariate polynomials* $c_1(u), \ldots, c_k(u) \in \mathbb{F}[u]$ *as follows*

$$c_i(u) \triangleq \prod_{j \in [k],\, j \neq i} \frac{u - \gamma_j}{\gamma_i - \gamma_j}$$

so that $c_i(\gamma_j)$ *is zero if* $i \neq j$ *and one if* $i = j$*. We define the function*

$$M : (\mathbb{F}^r)^k \times \mathbb{F} \mapsto \mathbb{F}^r$$

as follows:

$$M(x_1, \ldots, x_k, u) \triangleq \sum_{i=1}^{k} c_i(u) \cdot x_i.$$

Notice that, for a fixed $x = (x_1, \ldots, x_k)$, the output of M is indeed a curve (of degree $k-1$) that passes through each one of the points $x_1, \ldots, x_k$. The following

[3] To be more precise, we will need to refer to a *family* of extractors (or mergers) in order for the term 'polynomial time' to be meaningful. However, for the sake of brevity, and since these issues are standard, we will use the word 'explicit' for single functions.

theorem, showing the existence of good mergers, is the main result of this section. For convenience, the theorem is stated in terms of binary sources (sources over $\{0,1\}$ alphabet). This will allow us to choose the field size according to the required entropy of the output.

THEOREM 3.2. *For every $\alpha > 0$ and every n and $k \leq 2^{o(n)}$, there exists an explicit (m,ϵ)-merger $M : (\{0,1\}^n)^k \times \{0,1\}^d \mapsto \{0,1\}^n$ with*

$$m = (1-\alpha)\cdot n,$$

$$d = O(\log(n) + \log(k))$$

and

$$\epsilon = O\left((n\cdot k)^{-1}\right).$$

Proof. Let $\mathbb{F}$ be a finite field of size $q = 2^d$ such that

$$(n\cdot k)^{4/\alpha} < q \leq 2(n\cdot k)^{4/\alpha}.$$

We will assume w.l.o.g that $n = r\cdot d$ for some integer r (we can lose at most $d \leq o(n)$ number of bits of entropy this way but this is negligible). We can thus treat each block X_i as distributed over $\mathbb{F}^r$. Let

$$M : \left(\mathbb{F}^r\right)^k \times \mathbb{F} \mapsto \mathbb{F}^r$$

be given by Construction 3.1. We will show that M satisfies the requirements of the theorem.

First, notice that the seed length of M is

$$d = \log(q) = O(\log(n) + \log(k))$$

which is what we wanted. Let

$$\epsilon = 4\cdot q^{-\alpha/4}$$

and notice that indeed $\epsilon = O\left((n\cdot k)^{-1}\right)$. Let U denote a random variable uniform over $\mathbb{F} \simeq \{0,1\}^d$ and independent of X. Assume w.l.o.g that X_1 is uniform (the proof will be identical if another source is uniform). Let

$$Z = M(X,U)$$

denote the output of the merger. Using Lemma 5.4 from [11] we know that if Z is ϵ-far from having min entropy $(1-\alpha)n$ then there exists a set $T \subset \mathbb{F}^r$ of size

$$|T| \leq 2^{n(1-\alpha)} = q^{r(1-\alpha)}$$

such that

$$\Pr[Z \in T] \geq \epsilon.$$

Let

$$s = q^{1-\alpha/2},$$

and observe that, since $r < n < q^{\alpha/4}$, we have

$$\left(\frac{s}{r}\right)^r \geq q^{r(1-\alpha)} \geq |T|.$$

The expression on the left-hand-side is a lower bound on the number of monomials in r variables and degree at most s. Therefore, there are more monomials of degree $\leq s$ than points in T. We can thus find (by solving a system of linear equations) a non-zero degree $\leq s$ polynomial $g \in \mathbb{F}[y_1, \ldots, y_r]$ such that $g(y) = 0$ for all $y \in T$. Our goal is now to show that g is zero on many more points in $\mathbb{F}^r$, thus deriving a contradiction (since a low degree polynomial can't be zero on too many points). To show this, we will use the special structure of the output distribution of M.

For each $x_1 \in \mathbb{F}^r$ let

$$p_{x_1} \triangleq \Pr[Z \in T | X_1 = x_1]$$

and let

$$G \triangleq \{x_1 \in \mathbb{F}^r | p_{x_1} \geq \epsilon/2\}.$$

Then, by an averaging argument and since X_1 is uniform, we get

$$\Pr[X_1 \in G] = |G| \cdot q^{-r} \geq \epsilon/2.$$

Our contradiction will follow from the next claim, showing that g is zero on all points in G. The intuition for the proof is that for $x_1 \in G$, there exists a degree $k-1$ curve passing through x_1, that intersects T in 'too many' points. This, in turn, implies that the restriction of g to this curve (which is also a low degree polynomial) is identically zero and so $g(x_1) = 0$.

CLAIM 3.3. *Let $x_1 \in G$. Then $g(x_1) = 0$.*

Proof. Since the conditional probability of Z being in T given $X_1 = x_1$ is at least $\epsilon/2$ we can fix the random variables $X_2, \ldots, X_k$ to values $x_2, \ldots, x_k \in \mathbb{F}^r$ such that we still have

$$\Pr[Z \in T | X = (x_1, \ldots, x_k)] \geq \epsilon/2$$

(notice that in the above expression the only randomness comes from the seed). Let

$$C \triangleq \left\{ \sum_{i=1}^{k} c_i(u) \cdot x_i \middle| u \in \mathbb{F} \right\}.$$

Then, the restriction of g to C is given by the univariate polynomial

$$h(u) \triangleq g\left(c_1(u) \cdot x_1 + \ldots + c_k(u) \cdot x_k\right),$$

which has degree at most $s \cdot (k-1)$. From the above discussion, the polynomial $h(u)$ is zero on at least an $\epsilon/2$-fraction of $\mathbb{F}$ and since (using the bound $k < q^{\alpha/4}$)

$$\frac{s \cdot (k-1)}{q} < \frac{\epsilon}{2}$$

we get that, from Lemma 2.1, h must be zero on all points in $\mathbb{F}$ and in particular on $u = \gamma_1$ (where γ_1 is given by Construction 3.1). Therefore

$$0 = h(\gamma_1) = g(c_1(\gamma_1) \cdot x_1 + \ldots + c_k(\gamma_1) \cdot x_k) = g(x_1)$$

which is what we wanted to prove. □

We now get a contradiction to Lemma 2.1 since, by the above claim and by the bound on $|G|$, we get that g is zero on an $\epsilon/2$ fraction of the space $\mathbb{F}^r$ and this is a contradiction since $\epsilon/2 > s/q$. □

4. A merger-based extractor. In this section we show how to combine the Curve Merger of Section 3 with classical extractor machinery to derive an extractor which is optimal, up to constant factors, in all parameters (for inverse polynomial error). Besides the merger of Theorem 3.2, we will use the following result of Reingold, Shaltiel and Wigderson [20], which gives an optimal extractor for block soruces.

LEMMA 4.1 ([20]). *Let $n = n_1 + n_2$ and let k_1, k_2 be such that $k_2 > \log^4(n_1)$. Then there exists an explicit (k_1, k_2, ϵ)-block source extractor $E : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \times \{0,1\}^d \mapsto \{0,1\}^m$ with $m = k_1$, $d = O(\log(n))$ and $\epsilon = n^{-\Omega(1)}$.* We can use the above lemma to derive a simple extractor for sources with very high min entropy. This extractor will be useful to us since the output of the merger will only give us a source with high min entropy and our final goal is to get a source which is close to uniform.

COROLLARY 4.2. *For every $\alpha > 0$ the following holds: Suppose $k = (1-\alpha)n$. Then there exists an explicit (k, ϵ)-extractor $E : \{0,1\}^n \times \{0,1\}^d \mapsto \{0,1\}^m$ with $\epsilon = n^{-\Omega(1)}$, $d = O(\log(n))$ and $m = (1-3\alpha) \cdot n$.*

Proof. If we partition the source X into two blocks (X_1, X_2) of length $(1-2\alpha) \cdot n$ and $2\alpha \cdot n$ then we get a source which is $n^{-\Omega(1)}$-close to a (k_1, k_2)-block source with $k_2 > \Omega(n)$ and $k_1 \geq (1-3\alpha) \cdot n$ (see [12] for a proof of this fact). Applying the block source extractor from Lemma 4.1 gives us the required extractor. □

Another technical tool we will require is the following lemma which follows from the work of Ta-Shma [24]. This lemma gives a simple way to transform an arbitrary source into a somewhere block source.

LEMMA 4.3. *Let X be an (n,k)-source with $k > 10 \cdot \log^4(n)$. For each $t \in [n]$ let $Y^{(t)} = (Y_1^{(t)}, Y_2^{(t)})$ denote the partition of X into two consecutive blocks of length t and $n-t$. That is, $Y^{(t)}$ is distributed over $\{0,1\}^t \times \{0,1\}^{n-t}$. Let $Y = (Y^{(1)}, \ldots, Y^{(n)})$. Then Y is $(1/n)$-close to a somewhere (k_1, k_2)-block source, with $k_1 = k - 2\log^4(n)$ and $k_2 = \log^4(n)$.*

Proof. This follows from Lemma 2.3.1 in [24]. □

We will construct our extractor in four steps. The first will be to use Lemma 4.3 to convert the source into a somewhere block source. The second step will be to apply the block-source extractor from Lemma 4.1 on each of these blocks (with the same seed) to obtain a somewhere-random source. Then we will use the merger of Theorem 3.2 to merge these blocks into a single block which is close to having min entropy rate close to one. The final step will be to apply the extractor given by Corollary 4.2 to obtain a distribution which is close to uniform. The error in each one of these steps will be polynomially small (in n) and so we will get a polynomially small error for the entire construction.

THEOREM 4.4. *Suppose $k > \log^5(n)$ and let $\beta > 0$ be some constant. Then, there exists an explicit (k, ϵ)-extractor $E : \{0,1\}^n \times \{0,1\}^d \mapsto \{0,1\}^m$ with $m = (1-\beta) \cdot k$, $d = O(\log(n))$ and $\epsilon = n^{-\Omega(1)}$.*

Proof. Let X be an (n,k)-source with $k > 10 \cdot \log^4(n)$. For each $t \in [n]$ let $Y^{(t)} = (Y_1^{(t)}, Y_2^{(t)})$ be as in Lemma 4.3 and let $Y = (Y^{(1)}, \ldots, Y^{(n)})$. Let $k_1 = k - 2\log^4(n)$ and $k_2 = \log^4(n)$. For each $t \in [n]$ let

$$E_t : \left(\{0,1\}^t \times \{0,1\}^{n-t}\right) \times \{0,1\}^{d_1} \mapsto \{0,1\}^m$$

be the $(k_1, k_2, n^{-\Omega(1)})$-block source extractor given by Lemma 4.1, such that $d_1 = O(\log(n))$ and $m = k_1$. Let S_1 denote a random variable uniformly distributed over

$\{0,1\}^{d_1}$ and independent of X. For each $t \in [n]$ let us denote by

$$Z_t = E_t\left(Y^{(t)}, S_1\right)$$

and let $Z = (Z_1, \ldots, Z_n)$. Then, by Lemma 4.3 and Lemma 4.1 we have that Z is $n^{-\Omega(1)}$ close to a somewhere-random source with n blocks of length k_1-bits each. Notice that we can use the same seed S_1 in each one of the applications of E_t since the definition of a somewhere-random source allows for arbitrary dependencies among the different blocks.

Our next step is to apply the merger of Theorem 3.2 on Z. Let

$$M : \left(\{0,1\}^{k_1}\right)^n \times \{0,1\}^{d_2} \mapsto \{0,1\}^m$$

be the merger given by Theorem 3.2 with

$$d_2 = O(\log(k_1) + \log(n)) = O(\log(n))$$

and

$$m = (1 - \beta/10) \cdot k_1 > (1 - \beta/4) \cdot k.$$

Notice that the error incurred by this merger is $n^{-\Omega(1)}$. Let S_2 be a random variable distributed uniformly over $\{0,1\}^{d_2}$ and independent of X and of S_1 and let

$$W = M(Z, S_2)$$

denote the output of the merger on input Z and with seed S_2. Then, by Theorem 3.2, we have that W is $n^{-\Omega(1)}$-close to having min entropy at least $(1-\beta/10) \cdot k_1$. We can thus apply the extractor from Corollary 4.2 (with an additional independent seed of length $O(\log(n))$ bits) to get a source which is $n^{-\Omega(1)}$-close to uniform and has length $(1 - \beta) \cdot k$.

Notice that in each one of the four steps of the construction we used a seed which has length $O(\log(n))$ and so our total seed requirements are logarithmic in n, as was required. □

5. Extensions to the merger analysis.

5.1. The 'somewhere high-entropy' case. It is possible to consider, instead of somewhere-random sources, sources which have one block that has only high min entropy. It is then natural to try and construct mergers for such sources that preserve 'most' of the entropy of the 'good' block. In this section we show that the Curve Merger also works in this setting. The proof will be by direct reduction and will only rely on the fact that the merger commutes with any linear projection. Since we will not need to apply this merger in the binary setting we will state this result over the finite field $\mathbb{F}$. Making the transition to binary strings is standard and the field size can be chosen, as is done in Theorem 3.2, to control the statistical error.

We will say that a random variable $X \in (\mathbb{F}^n)^k$ is a somewhere s-source if there exists an index $i \in [k]$ such that the min entropy of X_i is at least $s \cdot \log(q)$ (one can also allow for convex combinations of such sources, as in Definition 2.3). Notice that s doesn't have to be an integer.

THEOREM 5.1. *Let $\alpha > 0$. Let $X = (X_1, \ldots, X_k) \in (\mathbb{F}^n)^k$ be a somewhere s-source, where s is bigger than some absolute constant C. Let $M : (\mathbb{F}^n)^k \times \mathbb{F} \mapsto \mathbb{F}^n$*

be defined as in Construction 3.1 and let Y be a random variable uniform on $\mathbb{F}$ and independent of X. Suppose $q > (s \cdot k)^{10/\alpha}$. Then, $M(X,Y)$ is ϵ-close to having min-entropy $\geq (1-\alpha) \cdot s \cdot \log(q)$, where $\epsilon = q^{-\Omega(1)}$.

Proof. Suppose that X_1 has min-entropy $\geq s \cdot \log(q)$ (the argument will be the same for other blocks) and let

$$m = \lceil (1 - \alpha/10) \cdot s \rceil.$$

A standard application of the Leftover Hash Lemma [14] shows that there exists a *linear* projection

$$\pi : \mathbb{F}^n \mapsto \mathbb{F}^m$$

such that $\pi(X_1)$ is δ-close to uniform, with $\delta = q^{-\Omega(1)}$. Let $Z = M(X,Y)$ and observe that

$$\pi(Z) = \pi(M(X,Y)) = M'(\pi(X_1), \ldots, \pi(X_k), Y)$$

where $M' : (\mathbb{F}^m)^k \times \mathbb{F} \mapsto \mathbb{F}^m$ is again given by Construction 3.1. Since $\pi(X)$ is δ-close to a somewhere-random source, and since the field size is large enough, we can carry on the analysis given in the proof of Theorem 3.2 and get that $\pi(Z)$ is $q^{-\Omega(1)}$-close to having min entropy $(1-\alpha) \cdot s \cdot \log(q)$ (we omit some simple calculations). The proof is now completed since applying a fixed function π cannot increase the min entropy and so Z is also close to having min entropy $(1-\alpha) \cdot s \cdot \log(q)$. □

We observe also that the same technique applied in the proof above (taking a random projection to a smaller space) can be used to give a bound on the size of Kakeya sets that contain only a 'few' lines. We do not include a full proof of this result since it is subsumed by the results in [9] (which appeared after the initial publication of this paper).

THEOREM 5.2. *For every $\epsilon > 0$ and every integer $s > 2$ there exists a constant $C_{s,\epsilon}$ such that if $K \subset \mathbb{F}^n$ contains lines in at least q^λ directions and $s = \lfloor \lambda \rfloor$ than*

$$|K| \geq C_{s,\epsilon} \cdot q^{\lambda - \epsilon}.$$

Proof. (Sketch) We project K onto $\mathbb{F}^{s-1}$ using a random projection $\pi : \mathbb{F}^n \mapsto \mathbb{F}^{s-1}$. Using the same argument as above we can deduce that there exists π such that $\pi(K)$ contains at least $(1/2) \cdot q^{s-1}$ lines (we count lines with multiplicities but this doesn't matter). We can now prove, as in [8], that $|\pi(K)| \geq C_s \cdot q^{s-1}$. Using tensoring (see Corollary 1.2 in [8]) this bound can be 'lifted' to $C_{s,\epsilon} \cdot q^{\lambda-\epsilon}$ for every $\epsilon > 0$. The constant $C_{s,\epsilon}$ can be seen to be of the order of $(s/\epsilon)^{-s}$. □

5.2. The computational setting. Here we show that our merger, indeed any efficient merger, performs as well when the input distribution is *computationally* somewhere random. To properly define this setting, we need to first define the type of distributions we will consider, and their "samplability". The samplers below explicitly generate "somewhere" distributions, by which we mean a distribution of several random variables of which one has some property, e.g being random, pseudorandom, having high min-entropy etc.

DEFINITION 5.3 (Somewhere sampler). *A function $S : \{0,1\}^n \times \{0,1\}^m \to (\{0,1\}^n)^k$ is a* somewhere-sampler *if there is an $i \in [k]$ such that for every input pair*

$x \in \{0,1\}^n$ and $r \in \{0,1\}^m$ we have $S(x,r)_i = x$. We call a sampler S efficient *if m is polynomial in n, and S is computable by a polynomial size circuit*[4].

Consider applying S to independent distributions X, R, where X is arbitrary and $R = U_m$. Then S is a probabilistic algorithm to convert a distribution X into a distribution on k-tuples where X is "hidden" in one of its components. In particular, S preserves the randomness content of X (eg if X has high min-entropy, then $S(X, U_m)$ is *somewhere* high min-entropy, and every high min-entropy distribution may be viewed this way. When S is efficient, a computational version analog holds, and for this we first define computational min-entropy.

DEFINITION 5.4 (Computational min-entropy). *Two distributions X, Z on $\{0,1\}^n$ are called* computationally indistinguishable *if for every polynomial size (distinguisher) circuit D we have $|\Pr[D(X) = 1] - \Pr[D(Z) = 1]| \leq n^{-\omega(1)}$. The* computational min-entropy *of a distribution X is at least k if it is computationally indistinguishable from some distribution Z with min-entropy at least k. When $k = n$ we call the distribution X* pseudorandom.

We note that the notion of computational min-entropy is quite subtle, and there is more than one natural definition for it. The one above is perhaps the most natural, and is taken from the seminal paper of Hastad et al [13]. We refer the reader to [4] for thorough discussion of these definitions and the interrelations between them (including many interesting open problems).

The simple observation of this section (for which we have no applications but hope it may find some), is that our merger, indeed any efficient merger, performs as well in this setting as in the information theoretic one, when the source is generated efficiently.

THEOREM 5.5. *Let $\alpha > 0$ and let M be the merger of Construction 3.1 with $|\mathbb{F}| > (nk)^{4/\alpha}$. Then for every distribution X with computational min-entropy s, every* efficient *somewhere sampler S whose randomness U_m is independent of X, we have that $M(S(X, U_m), Y)$ is $(1/nk)$-close to having computational min-entropy $(1-\alpha)s$, where Y is an independent variable distributed uniformly in $\mathbb{F}$.*

Proof. Take any distribution Z of min-entropy s, and consider $Z' = M(S(Z, U_m), Y)$. By our information theoretic merger Theorem 5.1 we know that Z' is $1/(nk)$-close to having min-entropy $(1-\alpha)s$. If the assertion of the theorem is wrong, then there exists an efficient distinguisher D which tells apart Z' from $M(S(X, U_m)Y)$. But then the composed (efficient) circuit $D \circ M \circ S$ tells apart Z from X, contradicting our assumption that X has computational min-entropy s.
□

6. Conclusions and Open Problems. In this work we designed mergers that are optimal up to constant factors in both seed length and output length. This led us (via known techniques) to a similarly optimal construction of seeded extractors. The next obvious challenges in the design of seeded extractors are

- Reducing the seed length to the optimal $(1 + o(1)) \cdot \log(n/\epsilon^2)$.
- Extracting all the entropy, rather than just a constant fraction (while keeping the seed logarithmic).

Both of these challenges make perfect sense when one is restricted to designing mergers, and we believe that attacking mergers first may be easier.

[4] As usual we should formally be talking about an ensemble of distributions and circuits, one for every n, etc.

Another challenge is to better understand the relation between our merger and the [11] condenser. Both use polynomials, in seemingly different ways, but we feel that the right viewpoint may render them more similar, or at least offer a common generalization, which would be of interest.

7. Acknowledgments. We thank Ran Raz and Amir Shpilka for helpful discussions. Research supported by Binational Science Foundation (BSF) grant and by NSF grant CCR-0324906.

REFERENCES

[1] A. Besicovitch. On Kakeya's problem and a similar one. *Mathematische Zeitschrift*, (27):312–320, 1928.
[2] J. Bourgain. On the dimension of Kakeya sets and related maximal inequalities. *Geom. Funct. Anal.*, (9):256–282, 1999.
[3] J. Bourgain. Harmonic analysis and combinatorics: How much may they contribute to each other? *IMU/Amer. Math. Soc.*, pages 13–32, 2000.
[4] B. Barak, R. Shaltiel, and A. Wigderson. Computational analogues of entropy. In *11th International Conference on Random Structures and Algorithms*, pages 200–215, 2003.
[5] B. Chor and O. Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. *SIAM Journal on Computing*, 17(2):230–261, April 1988. Special issue on cryptography.
[6] Z. Dvir, S. Kopparty, S. Saraf, and M. Sudan. Extensions to the method of multiplicities, with applications to kakeya sets and mergers. In FOCS 09 (to appear), 2009.
[7] Z. Dvir and A. Shpilka. An improved analysis of linear mergers. *Comput. Complex.*, 16(1):34–59, 2007. (Extended abstract appeared in RANDOM 2005).
[8] Z. Dvir. On the size of Kakeya sets in finite fields. *J. AMS (to appear)*, 2008.
[9] J. Ellenberg, R. Oberlin, and T. Tao. The Kakeya set and maximal conjectures for algebraic varieties over finite fields, 2009. Manuscript.
[10] C. Fefferman. The multiplier problem for the ball. *Annals of Mathematics*, (94):330–336, 1971.
[11] V. Guruswami, C. Umans, and S. Vadhan. Unbalanced expanders and randomness extractors from parvaresh-vardy codes. In *CCC '07: Proceedings of the Twenty-Second Annual IEEE Conference on Computational Complexity*, pages 96–108, Washington, DC, USA, 2007. IEEE Computer Society.
[12] Oded Goldreich and Avi Wigderson. Tiny families of functions with random properties: A quality-size trade-off for hashing. *Random Structures & Algorithms*, 11(4):315–343, 1997.
[13] J. HÅstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.
[14] R. Impagliazzo, L.A. Levin, and M. Luby. Pseudorandom generation from one-way functions. In *Proceedings of the 21st ACM Symposium on Theory of Computing*, 1989.
[15] Nets Katz and Terence Tao. Recent progress on the Kakeya conjecture. In *Publicacions Matematiques, Proceedings of the 6th International Conference on Harmonic Analysis and Partial Differential Equations*, pages 161–180, U. Barcelona, 2002.
[16] Chi-Jen Lu, Omer Reingold, Salil Vadhan, and Avi Wigderson. Extractors: Optimal up to constant factors. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing"*, 2003.
[17] G. Mockenhaupt and T. Tao. Restriction and Kakeya phenomena for finite fields. *Duke Math. J.*, 121:35–74, 2004.
[18] Farzad Parvaresh and Alexander Vardy. Correcting errors beyond the guruswami-sudan radius in polynomial time. In *FOCS '05: Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 285–294, Washington, DC, USA, 2005. IEEE Computer Society.
[19] K.M Rogers. The finite field Kakeya problem. *Amer. Math. Monthly 108*, (8):756–759, 2001.
[20] O. Reingold, R. Shaltiel, and A. Wigderson. Extracting randomness via repeated condensing. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, 2000.
[21] R. Shaltiel. Recent developments in explicit constructions of extractors. *Bulletin of the EATCS*, 77:67–95, 2002.
[22] Ronen Shaltiel and Christopher Umans. Simple extractors for all min-entropies and a new pseudorandom generator. *J. ACM*, 52(2):172–216, 2005.

[23] T. Tao. From rotating needles to stability of waves: emerging connections between combinatorics, analysis, and pde. *Notices Amer. Math. Soc.*, 48(3):294–303, 2001.
[24] A. Ta-Shma. *Refining Randomness*. PhD thesis, The Hebrew Univerity, Jerusalem, Israel, 1996.
[25] T. Wolff. Recent work connected with the Kakeya problem. pages 129–162, 1999.