Electronic Colloquium on Computational Complexity, Report No. 161 (2011)

Design Extractors, Non-Malleable Condensers and Privacy Amplification

Xin Li*
Department of Computer Science
University of Washington
Seattle, WA 98905, U.S.A.
lixints@cs.washington.edu

Abstract

We introduce a new combinatorial object, called a *design extractor*, that has both the properties of a design and an extractor. We give efficient constructions of such objects and show that they can be used in several applications.

1. **Improving the output length of known non-malleable extractors.** Non-malleable extractors were introduced in [DW09] to study the problem of privacy amplification with an active adversary. Currently, only two explicit constructions are known [DLWZ11, CRS11]. Both constructions work for n bit sources with min-entropy $k > n/2$. However, in both constructions the output length is smaller than the seed length, while the probabilistic method shows that to achieve error ϵ, one can use $O(\log n + \log(1/\epsilon))$ bits to extract up to $k/2$ output bits. In this paper, we use our design extractor to give an explicit non-malleable extractor for min-entropy $k > n/2$, that has seed length $O(\log n + \log(1/\epsilon))$ and output length $\Omega(k)$.
2. **Non-malleable condensers.** We introduce and define the notion of a *non-malleable condenser*. A non-malleable condenser is a generalization and relaxation of a non-malleable extractor. We show that similar as extractors and condensers, non-malleable condensers can be used to construct non-malleable extractors. We then show that our design extractor already gives a non-malleable condenser for min-entropy $k > n/2$, with error ϵ and seed length $O(\log(1/\epsilon))$.
3. **A new optimal protocol for privacy amplification.** More surprisingly, we show that non-malleable condensers themselves give optimal privacy amplification protocols with an active adversary. In fact, the non-malleable condensers used in these protocols are much weaker compared to non-malleable extractors, in the sense that the entropy rate of the condenser's output does not need to increase at all. This suggests that one promising next step to achieve better privacy amplification protocols may be to construct non-malleable condensers for smaller min-entropy. As a by-product, we also obtain a new explicit 2-round privacy amplification protocol with optimal entropy loss and optimal communication complexity for min-entropy $k > n/2$, without using non-malleable extractors.

*Partially supported by NSF Grants CCF-0634811, CCF-0916160, THECB ARP Grant 003658-0113-2007, and a Simons postdoctoral fellowship.

ISSN 1433-8092

1 Introduction

Over the past decades, motivated by the problem of using imperfect random sources in computation, seeded randomness extractors [NZ96] have been studied extensively [SZ99, Tre01, RRV02, LRVW03, GUV09, DW08, DKSS09]. Besides its original motivation, seeded extractors have also found applications in cryptography, coding theory, complexity theory and many other areas. When viewed as a bipartite graph, seeded extractors are closely related to expander graphs, super concentrators [WZ99] and randomness optimal samplers [Zuc97]. Today we have seeded extractors with nearly optimal parameters [LRVW03, GUV09, DW08, DKSS09].

Another combinatorial object, known as design, has also been widely used in computer science. For example, it has been used in the construction of the Nisan-Wigderson pseudorandom generator [NW94] and the construction of Trevisan's extractor [Tre01]. Previously, except the use of designs in Trevisan's extractor, there is no known connection between these two objects.

In this paper, we introduce a new combinatorial object called *design extractor*. As evident from the name, it is both a design and an extractor. In other words, when viewed as a bipartite graph, we require this object to have both the properties of a design and an extractor. Informally, the two properties are:

- For any subset S of vertices on the right with density ρ, most of the vertices on the left have roughly a ρ fraction of neighbors in S. This is the extractor property.
- For any two different vertices on the left, the intersection of their neighbors on the right has a small size (compared to the left degree). This is the design property. Note that we are viewing the neighbors of left vertices as subsets of right vertices.

Formally, we have the following definition.

Definition 1.1. An $(N, M, K, D, \alpha, \epsilon)$ design extractor is a bipartite graph with left hand side $[N]$, right hand side $[M]$, left degree D such that the following properties hold.

- (extractor property) For any subset $S \subseteq [M]$, let $\rho_S = |S|/M$. For any vertex $v \in [N]$, let $\rho_v = |\Gamma(v) \cap S|/D$. Let $Bad_S = \{v \in [N] : |\rho_v - \rho_S| > \epsilon\}$, then $|Bad_S| \leq K$.
- (design property) For any two different vertices $u, v \in [N]$, $|\Gamma(u) \cap \Gamma(v)| \leq \alpha D$.

At first it may not seem obvious that such objects exist. However we show that design extractors exist with good parameters, and in fact they can be constructed very naturally from seeded extractors. We also give much more efficient constructions of design extractors based on a variant of Trevisan's extractor. We then show that design extractors are useful in several applications.

1.1 Improving the output length of known non-malleable extractors

Non-malleable extractors were introduced by Dodis and Wichs [DW09] to give protocols for the problem of privacy amplification with an active adversary. We now give the formal definition below.

Notation. We let $[s]$ denote the set $\{1, 2, \ldots, s\}$. For ℓ a positive integer, U_ℓ denotes the uniform distribution on $\{0,1\}^\ell$, and for S a set, U_S denotes the uniform distribution on S. When used as a component in a vector, each U_ℓ or U_S is assumed independent of the other components. We say $W \approx_\varepsilon Z$ if the random variables W and Z have distributions which are ε-close in variation distance.

Definition 1.2. The *min-entropy* of a random variable X is

$$H_\infty(X) = \min_{x \in \mathsf{supp}(X)} \log_2(1/\Pr[X = x]).$$

For $X \in \{0,1\}^n$, we call X an $(n, H_\infty(X))$-source, and we say X has *entropy rate* $H_\infty(X)/n$.

Definition 1.3. A function $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is a *strong* (k,ε)*-extractor* if for every source X with min-entropy k and independent Y which is uniform on $\{0,1\}^d$,

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

Definition 1.4. A function $\mathsf{nmExt} : [N] \times [D] \to [M]$ is a (k,ε)-non-malleable extractor if, for any source X with $H_\infty(X) \geq k$ and any function $\mathcal{A} : [D] \to [D]$ such that $\mathcal{A}(y) \neq y$ for all y, the following holds. When Y is chosen uniformly from $[D]$ and independent of X,

$$(\mathsf{nmExt}(X,Y), \mathsf{nmExt}(X,\mathcal{A}(Y)), Y) \approx_\varepsilon (U_{[M]}, \mathsf{nmExt}(X,\mathcal{A}(Y)), Y).$$

Remark 1.5. Dodis and Wichs originally defined *average-case non-malleable extractor*, while our definition here is *worst-case non-malleable extractor*. However, these two notions are essentially equivalent, up to a small change of parameters. Throughout the rest of our paper, when we say non-malleable extractor, we refer to the worst-case non-malleable extractor of Definition 1.4.

In [DW09], using the probabilistic method, Dodis and Wichs showed that non-malleable extractors exist when $k > 2m + 3\log(1/\varepsilon) + \log d + 9$ and $d > \log(n-k+1) + 2\log(1/\varepsilon) + 7$, for $N = 2^n$, $M = 2^m$, and $D = 2^d$. However, no explicit constructions of non-malleable were given in [DW09].

The first explicit non-malleable extractor was given in [DLWZ11], where the authors constructed a non-malleable extractor that works for $k > n/2$. Using this construction, they achieved a 2-round privacy amplification protocol with optimal entropy loss for $k > n/2$. However, one drawback is that their result uses a large seed length $d = n$ (although this restriction was later removed) and their efficiency when outputting more than $\log n$ bits relies on an unproven assumption. Later, Cohen, Raz, and Segev [CRS11] gave an alternative construction of a non-malleable extractor that also works for $k > n/2$. Their construction improves the result of [DLWZ11] in the sense that it works for any seed length d with $2.01\log n \leq d \leq n$, and does not rely on any unproven assumption. Thus the result of [CRS11] also improves the communication complexity of the protocols in [DLWZ11].

However, both the construction in [DLWZ11] and the construction [CRS11] suffer from another drawback: the output length is smaller than the seed length. Indeed, they only achieve $m = \alpha d$ for some constant $\alpha < 1$. Thus, if one wants a large output length, say $\Omega(n)$, then the seed length is also forced to be large. On the other hand, the probabilistic method shows that one can use roughly $O(\log n)$ bits to extract up to roughly $k/2$ bits. For $k > n/2$ this is $\Omega(n)$ bits. Since one of the ultimate goals in constructing randomness extractors is to use a small seed length to extract almost all the entropy in a weak source, here one can ask the natural question of whether we can improve the output length of the above constructions.

In this paper, we give a positive answer by using our design extractor. Specifically, we have the following theorem.

Theorem 1.6. *For every constant $\delta > 0$, there exists a constant $\beta > 0$ such that for every $n, k \in \mathbb{N}$ with $k \geq (1/2 + \delta)n$ and $\epsilon > 2^{-\beta n}$ there exists an explicit (k,ϵ) non-malleable extractor with seed length $d = O(\log n + \log \epsilon^{-1})$ and output length $m = \Omega(n)$.*

1.2 Non-malleable condensers

Next, we introduce the notion of a *non-malleable condenser*. Similar as the relation between extractors and condensers, a non-malleable condenser is a relaxation and generalization of a non-malleable extractor. Informally, given a weak source X, an independent seed Y, and a deterministic (adversarial) function $\mathcal{A}$ such that $\forall y, \mathcal{A}(y) \neq y$, we require that for most seeds y, with high probability over the fixing of the condenser's output on $\mathcal{A}(y)$, the condenser's output on y is close to having a certain amount of min-entropy. More formally, we have the following definition.

Definition 1.7. A (k, k', ϵ) non-malleable condenser is a function $\mathsf{nmCond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ such that given any (n,k)-source X, an independent uniform seed $Y \in \{0,1\}^d$, and any (deterministic) function $\mathcal{A} : \{0,1\}^d \to \{0,1\}^d$ such that $\forall y, \mathcal{A}(y) \neq y$, we have that with probability $1-\epsilon$ over the fixing of $Y = y$,

$$\Pr_{z' \leftarrow \mathsf{nmCond}(X, \mathcal{A}(y))}[\mathsf{nmCond}(X,y)|_{\mathsf{nmCond}(X,\mathcal{A}(y))=z'} \text{ is } \epsilon - \text{close to an } (m,k') \text{ source}] \geq 1-\epsilon.$$

It is easy to see that a non-malleable extractor is indeed a special case of a non-malleable condenser. We then show that our design extractor gives a natural construction of a non-malleable condenser for weak sources with min-entropy $k > n/2$. Specifically, we have

Theorem 1.8. *For any constant $\delta > 0$, there exists a constant $\beta > 0$ such that for any $n, k \in \mathbb{N}$ with $k \geq (1/2+\delta)n$ and $\epsilon > 2^{-\beta n}$, there exists an efficiently computable (k, k', ϵ) non-malleable condenser with $d = O(\log(1/\epsilon)), m = \Omega(n)$ and $k' \geq \delta m/2$.*

Note that the seed length here does not depend on n. We further show that similar as a condenser can be used to construct an extractor, a non-malleable condenser can also be used to construct a non-malleable extractor. For example, we have the following theorem.

Theorem 1.9. *Assume we have an explicit (k, k', ϵ) non-malleable condenser $\mathsf{nmCond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ such that $k' \geq (1/2+\delta)m$ for some constant $\delta > 0$ and $2^d \geq 1/\epsilon$. Then there exists an efficiently computable $(k, 9\epsilon)$ non-malleable extractor with seed length $O(d + \log m + \log \epsilon^{-1})$ and output length $\Omega(m)$.*

1.3 A new optimal privacy amplification protocol

Non-malleable extractors were first defined to give privacy amplification protocols in the presence of an active adversary. The basic setting is that, two parties Alice and Bob share a private weakly random string, and they wish to communicate with each other through a public channel controlled by an adversary with unlimited computational power. Assuming that both parties also have access to local private uniform random bits, the goal is that Alice and Bob should agree on a shared private nearly uniform string at the end of the protocol. The adversary is active in the sense that he/she can arbitrarily insert, delete, modify, and reorder messages, or even run several rounds with one party before resuming execution with the other party.

This problem has been studied by many researchers. Maurer and Wolf [MW97] gave the first non-trivial protocol for this problem, which works when the entropy rate of the weakly-random secret is bigger than 2/3. This was later improved by Dodis, Katz, Reyzin, and Smith [DKRS06] to work for entropy rate bigger than 1/2. Yet both protocols suffer a significant entropy loss. In

[DW09], Dodis and Wichs showed that there is no one-round protocol for entropy rate less than 1/2. Renner and Wolf [RW03] gave the first protocol for entropy rate below 1/2, and their result was simplified by Kanukurthi and Reyzin [KR09], who showed that the protocol can run in $O(s)$ rounds and achieve entropy loss $O(s^2)$ to achieve security parameter s.

In [DW09], Dodis and Wichs further showed that explicit non-malleable extractors give optimal 2-round protocols for the problem of privacy amplification with an active adversary, and these protocols also achieve asymptotically optimal entropy loss. However, they didn't give explicit non-malleable extractors and only gave a weaker form of non-malleable extractors called "look ahead" extractors. They used it to give a 2-round privacy amplification protocol with entropy loss $O(s^2)$. Chandran, Kanukurthi, Ostrovsky, and Reyzin [CKOR10] improved the entropy loss to $O(s)$ but the number of rounds is also $O(s)$. In [DLWZ11] and later in [CRS11], by giving the first explicit non-malleable extractors for min-entropy $k > n/2$, the authors obtained a 2-round privacy amplification protocol with optimal entropy loss $O(s)$ for $k > n/2$. [DLWZ11] also gave a constant round protocol with optimal entropy loss for min-entropy $k = \delta n$ with any constant $\delta > 0$.

An active adversary seems quite powerful and thus it seems natural that in order to get an optimal protocol, one needs some strong tools such as non-malleable extractors. However, one may still ask the question of whether a non-malleable extractor is truly needed to construct an optimal privacy amplification protocol. Surprisingly, we show that this is not the case. In fact, we show that a non-malleable condenser itself suffices to give a 2-round privacy amplification protocol with optimal entropy loss. Specifically, we have

Theorem 1.10. *There exists a constant $c > 1$ such that for any $n, k \in \mathbb{N}$ and $\epsilon > 0$, assume that we have an explicit (k, k', ϵ)-non-malleable condenser with $k' \geq c(\log n + \log(1/\epsilon))$ and $d = O(\log n + \log(1/\epsilon))$. Then there is an explicit 2-round privacy amplification protocol with security parameter $\log(1/\epsilon)$, entropy loss $O(\log n + \log(1/\epsilon))$ and communication complexity $O(\log n + \log(1/\epsilon))$.*

The existence of such non-malleable condensers follow directly from the existence of non-malleable extractors and the fact that a non-malleable extractor is also a non-malleable condenser. Note that in this theorem, there is no requirement on the min-entropy rate of the condenser's output. This means that it can even not condense the source in the traditional sense at all. Indeed it can even lower the min-entropy rate. All that is required is that the output on y has $c(\log n + \log(1/\epsilon))$ min-entropy conditioned on the output on $\mathcal{A}(y)$. Thus we see that this is a much weaker requirement than that of a non-malleable extractor. This, together with the fact that non-malleable condensers can be used to construct non-malleable extractors, suggest that one next and promising step to achieve better privacy amplification protocols may be to construct explicit non-malleable condensers for smaller min-entropy.

Plugging in our non-malleable condenser for min-entropy $k > n/2$, we also obtain a new 2-round privacy amplification protocol with optimal entropy loss and optimal communication complexity for min-entropy $k > n/2$, without using non-malleable extractors.

2 Overview of The Constructions and Techniques

In this section we give an overview of our constructions and the techniques used. In order to give a clean description, we shall be informal and imprecise sometimes. We start with the constructions of design extractors.

2.1 Constructions of design extractors

We first show that a design extractor can be constructed easily from a seeded extractor. Take a seeded extractor and view it as an (N, M, K, D) bipartite graph with N vertices on the left, M vertices on the right and left degree D. The extractor property says that for any subset S on the right with density $\rho = |S|/M$, most vertices on the left have roughly a ρ fraction of neighbors in S. Indeed, the number of vertices on the left that deviate from this property is at most K. Now consider a vertex x on the left. The neighbors of x is a subset S_x of the vertices on the right, with density $\rho_x = D/M$. Thus again by the extractor property, most vertices on the left will have roughly a ρ_x fraction of neighbors in S_x. This means that except for $K-1$ different vertices on the left, all the other vertices $y, y \neq x$ on the left will satisfy that $|\Gamma(y) \cap \Gamma(x)|$ is roughly $D/M \cdot D$. When D is small this is much smaller than D.

Now we can get a design extractor by the following greedy approach: first pick a vertex x on the left and get rid of all the other vertices on the left who have more than $D/M \cdot D$ neighbors in S_x, and then continue. Since each time we get rid of at most $K-1$ vertices, we are left with N/K vertices on the left, and now we get a design extractor.

One drawback of the above approach is that it takes time polynomial in N. Sometimes this is not good enough and we need the running time to be poly-logarithmic in N. Our next improvement comes from the observation that a certain kind of existing extractor construction is already a design extractor. This is simply Trevisan's extractor [Tre01]. To see this, note that for a strong extractor, if we add the seed to the output and then view it as a bipartite graph (which is still an extractor), we have that if two different vertices (two inputs) on the left have the same neighbor on the right, then this neighbor must come from the same seed of the two inputs. Note that Trevisan's extractor uses a binary code to encode the source, thus for two different inputs their encodings have a large hamming distance. Now consider the first bit of the output. It is selected from the bits of the encoded source by a subset of the bits from the seed. When the seed cycles over all possibilities, this subset also uniformly cycles over all bits of the encoded source. Thus for a large fraction of the seeds the outputs of two different inputs will be different. Therefore their neighbors will differ by a large fraction. Thus we already have a design extractor.

However, since Trevisan's extractor uses a binary code, its relative distance cannot be bigger than $1/2$. Sometimes we need a larger distance for the design extractor, thus we generalize Trevisan's extractor to use a code over a larger alphabet. We show that by using certain concatenated codes from [GS00] with good list-decoding bounds and large distance, the generalized extractor is still a strong extractor. Thus we get design extractors such that the intersection between different subsets is smaller.

2.2 Improving the output length of non-malleable extractors

Here our approach is similar to the block-source extractor used in [NZ96]. Specifically, we first view the extractor graph as a sampler, as in [Zuc97]. We note that by adding the seed to the output of a strong extractor and viewing it as a bipartite graph, the neighbors of any left vertex are all distinct. Thus we have samplers with distinct samples. Given any (n, k)-source X with $k = (1/2 + \delta)n$, we first use the sampler to sample a block with $\Omega(n)$ bits from the weak source. We limit the block length to be smaller than δn, so that even conditioned on the block, the original source X still has entropy rate $> 1/2$. We then take a small seed and apply a known non-malleable extractor to X and output another small seed R, which is independent of the sampled block. Finally we

use R as a seed and apply an optimal strong seeded extractor to the sampled block, thus we can output $\Omega(n)$ bits. To argue about the non-malleability, we need the outputs of the sampler to be a non-malleable condenser, and this follows from the design extractor property.

More specifically, assume that the error of the non-malleable extractor we are seeking is ϵ, we will construct a design extractor with $N = \text{poly}(1/\epsilon) = \text{poly}(K)$, $M = n$ and $D = \frac{\delta}{10}n$ such that the intersection of the neighbors of two different left vertices has size at most $\frac{\delta}{8}D$. We now associate the right vertices of the design extractor with the n bits of X, and choose a random seed Y_1 with $d_1 = \log N$ bits to sample $\bar{X}$ with D bits from X, according to the sampler. Next we take the non-malleable extractor nmExt from [CRS11] and use another random seed Y_2 with $d_2 = O(\log n + \log(1/\epsilon))$ bits to extract R from X, and output $d_3 = O(\log n + \log(1/\epsilon))$ bits. Now take any strong seeded extractor Ext with optimal parameters, for example the construction in [GUV09], and the final output is $Z = \mathsf{Ext}(\bar{X}, R)$ that outputs $\Omega(D) = \Omega(n)$ bits.

In the following discussion we use letters with prime to denote results obtained by computing with $\mathcal{A}(Y)$. The idea is as follows. When we use a sampler with distinct samples to sample $\bar{X}$ from the source X, by the analysis in [Vad04], for most seeds y_1 (indeed, except K y_1's), $\bar{X}$ roughly has min-entropy rate $1/2 + \delta$. We call these y_1's good. We now want to show that for most good seeds y_1, for any different $y_1' \neq y_1$, $\bar{X}$ sampled with y_1 has min-entropy δD conditioned on $\bar{X}$ sampled with y_1' (call it $\bar{X}'$). This can be shown as follows. First, if y_1' is good, then by similar analysis as in [Vad04] and the fact that $\bar{X}$ and $\bar{X}'$ have at most $\frac{\delta}{8}D$ bits in common, the joint distribution of $(\bar{X}, \bar{X}')$ has min-entropy roughly $2(1/2+\delta)D - \frac{\delta}{8}D > (1+1.5\delta)D$ (think of this as subtracting the entropy of the repeated bits). Since this entropy is larger than the length of $\bar{X}'$, which is D, we have that conditioned on $\bar{X}'$, $\bar{X}$ has min-entropy δD. Next, if y_1' is bad, since $\bar{X}'$ has only D bits, even conditioned on $\bar{X}'$, X still has min-entropy roughly $(1/2+0.9\delta)n$. Thus again by the analysis in [Vad04], except for K y_1's, $\bar{X}$ will roughly has min-entropy $(1/2+0.9\delta)D > \delta D$. In other words, a bad y_1' can ruin at most K good y_1's. Since there are at most K bad y_1''s, they can ruin at most K^2 good y_1's. Therefore, as long as $N >> K^2$, for most good seeds y_1, for any different $y_1' \neq y_1$, $\bar{X}$ sampled with y_1 has min-entropy δD conditioned on $\bar{X}$ sampled with y_1'.

Note that the above analysis already shows that this is a non-malleable condenser as we defined. Now note that the seed $Y = (Y_1, Y_2)$. To argue the non-malleability of our construction, we mainly use properties of block source extractors and the known non-malleable extractors. Take some $Y' = (Y_1', Y_2') \neq Y$, for now we assume that Y_1' is a deterministic function of Y_1 and Y_2' is a deterministic function of Y_2. We basically have the following three cases.

First, $Y_1 = Y_1'$ but $Y_2 \neq Y_2'$. The basic idea is that, $\bar{X}$ has min-entropy rate roughly $1/2 + \delta$. Note that even conditioned on $\bar{X}$, X still has min-entropy roughly $(1/2 + 0.9\delta)n$. Thus when we apply the non-malleable extractor in [CRS11] to X and Y_2, we get an output R that is uniform and independent of $\bar{X}$ and R' (the output of the non-malleable extractor applied to X and Y_2'). Therefore as long as the size of Z is not too large (but still $\Omega(D)$), we can fix $Z' = \mathsf{Ext}(\bar{X}, R')$ and $\bar{X}$ still has a constant fraction of min-entropy. Now we see $Z = \mathsf{Ext}(\bar{X}, R)$ is independent of Z'.

Second, $Y_1 \neq Y_1'$ but $Y_2 = Y_2'$. The basic idea is that, by above we know that even conditioned on $\bar{X}'$, $\bar{X}$ has has min-entropy δD. We also know that even if conditioned on $(\bar{X}, \bar{X}')$, X still has min-entropy roughly $(1/2+0.8\delta)n$ (since D is small). Thus we get that R is independent of $(\bar{X}, \bar{X}')$. Therefore by the property of a strong extractor, $Z = \mathsf{Ext}(\bar{X}, R)$ is independent of $Z' = \mathsf{Ext}(\bar{X}', R)$ and has $\Omega(n)$ bits.

Third, $Y_1 \neq Y_1'$ and $Y_2 \neq Y_2'$. Again, even conditioned on $\bar{X}'$, $\bar{X}$ has min-entropy δD. Also, even if conditioned on $(\bar{X}, \bar{X}')$, X still has min-entropy roughly $(1/2 + 0.8\delta)n$. Thus by the property of

the non-malleable extractor, R is independent of $(R', \bar{X}, \bar{X}')$. Therefore, we can now fix R' and $\bar{X}'$ (so Z' is fixed), and see that $Z = \mathsf{Ext}(\bar{X}, R)$ is uniform and has $\Omega(n)$ bits.

One subtle problem with the above argument is that it is not necessarily true that Y_1' is a deterministic function of Y_1 and Y_2' is a deterministic function of Y_2. In fact, Y_1', Y_2' can depend on both Y_1 and Y_2, thus $\bar{X}$ may be correlated with R through $\bar{X}'$. We can solve this by modifying our argument a little bit as follows. We first fix Y_1 and Y_1'. After this fixing Y_2' is indeed a deterministic function of Y_2. If $Y_1 = Y_1'$ then we can proceed as before. If $Y_1 \neq Y_1'$, then first, by the non-malleable condenser property for most choices of Y_1, no matter what Y_1' is , $\bar{X}$ has has min-entropy δD conditioned on $\bar{X}'$. Second, after this fixing, $(\bar{X}, \bar{X}')$ is a deterministic function of X, and are thus independent of Y_2 as well as R. However, since Y_1' may be a function of Y_2, fixing Y_1' may cause Y_2 to lose entropy. Luckily, it is shown in [DLWZ11] that a non-malleable extractor with uniform seed is also a non-malleable extractor with weak random seed, as long as the entropy loss is not too big compared to the seed length. Thus we can take the size of Y_2 to be a constant times bigger than the size of Y_1, and then fixing Y_1' won't cause Y_2 to lose much entropy. Now our argument can proceed as before.

Note that the error that an adversary can achieve is at most the sum of the errors in the above three cases. Thus we obtain a non-malleable extractor.

2.3 Constructing non-malleable extractor from non-malleable condenser

The construction is simple. Given any weak source X, assume we have a non-malleable condenser nmCond and a known non-malleable extractor nmExt. Just take two independent uniform random seeds Y_1, Y_2 and the output is $Z = \mathsf{nmExt}(\mathsf{nmCond}(X, Y_1), Y_2)$.

However, the analysis is not as straightforward as in the case of traditional condensers and extractors. It is actually similar to the analysis of improving the output length of a non-malleable extractor above. Given some $Y' = (Y_1', Y_2') \neq Y$, here we basically have two cases: $Y_1 = Y_1'$ or $Y_1 \neq Y_1'$. We also need the size of Y_2 to be larger than the size of Y_1 and use the property that a non-malleable extractor with uniform seed is also a non-malleable extractor with weak seed.

2.4 A new optimal privacy amplification protocol

Here our protocol is a generalization and modification of the protocol with a non-malleable extractor. So we first describe that protocol due to Dodis and Wichs [DW09]. The protocol also uses an additional tool known as a one-time message authentication code (MAC). Roughly speaking, a MAC authenticates a message m by using a private uniformly random key R to produce a tag T for the message, such that when an adversary does not know the key, the probability that he/she can guess the correct tag T' for another message $m' \neq m$ is small, even given m and T. We note that there are explicit constructions of MACs that work even if the adversary knows some partial information of the key R, namely as long as the entropy rate of R is bigger than $1/2$. We call this kind of MACs leakage-resilient MACs.

Now assume that we have a non-malleable extractor nmExt. Dodis and Wichs' protocol proceeds as follows. In the first round Alice chooses a fresh random string Y from her local random bits and sends it to Bob. Bob receives a possibly modified string Y'. They then compute $R = \mathsf{nmExt}(X, Y)$ and $R' = \mathsf{nmExt}(X, Y')$ respectively. Next, Bob chooses a fresh random string W' from his local random bits and sends it to Alice, together with a tag $T' = \mathsf{MAC}_{R'}(W')$ by using R' as the MAC key. Alice receives a possibly modified version (W, T), and she checks if $T = \mathsf{MAC}_R(W)$. If not, then

Alice rejects; otherwise they compute outputs $Z = \mathsf{Ext}(X, W)$ and $Z' = \mathsf{Ext}(X, W')$ respectively, where Ext is a seeded strong extractor.

The analysis of the above protocol is simple. If the adversary Eve does not change Y, then $R = R'$ and is (close to) uniform. Therefore the MAC ensures that the probability that Eve can change W' without being detected is very small. On the other hand if Eve changes Y, then the property of the non-malleable extractor guarantees that R' is independent of R. Thus in this case, again only with very small probability can Eve change W' without being detected. Now if $W = W'$, then Alice and Bob agree on a private uniform string by the property of the strong extractor.

Now we describe our protocol. In the first round, as above, we also have Alice choose a fresh random string Y_1 from her local random bits and send it to Bob, and they compute $R = \mathsf{Ext}(X, Y_1)$ and $R' = \mathsf{Ext}(X, Y_1')$ respectively, where Ext is a (traditional) seeded strong extractor (note that now we don't have a non-malleable extractor). In the next round, Bob chooses a fresh random string W' from his local random bits and sends it to Alice, together with a tag $T_2' = \mathsf{MAC}_{R'}(W')$ by using R' as the MAC key. Alice receives a possibly modified version (W, T_2), and she checks if $T_2 = \mathsf{MAC}_R(W)$. If not, then Alice rejects; otherwise they compute outputs $Z = \mathsf{Ext}(X, W)$ and $Z' = \mathsf{Ext}(X, W')$ respectively.

The above protocol guarantees that in the case where the adversary does not change Y_1, W' can be authenticated to Alice and thus the two parties can agree on a private uniform random string in the end. We now consider the case where the adversary does change Y_1. Assume now that we have a non-malleable condenser nmCond. We modify the above protocol such that at the end of the first round, Alice and Bob also compute $Z = \mathsf{nmCond}(X, Y_1)$ and $Z' = \mathsf{nmCond}(X, Y_1')$ respectively. Note that the output of the non-malleable condenser may not be uniform, in fact it may not even have high min-entropy rate. Thus we cannot use Z or Z' as a MAC key. However, we still need to use the property of the non-malleable condenser to ensure that, when Eve does change Y_1, the probability that she can change W' without being detected is very small. To this end, in the first round we have Alice choose another fresh random string Y_2 and also send it to Bob. Bob receives a possibly modified version Y_2'. Now Alice computes another tag $\bar{T}_1 = \mathsf{Ext}(Z, Y_2)$ and in the second round, Bob also computes $T_1' = \mathsf{Ext}(Z', Y_2')$ and sends it to Alice, where Alice may receive a modified version T_1. Now Alice will check two tags. If either $\bar{T}_1 \neq T_1$ or $\mathsf{MAC}_R(W) \neq T_2$ she rejects; otherwise they compute outputs $Z = \mathsf{Ext}(X, W)$ and $Z' = \mathsf{Ext}(X, W')$ respectively.

The basic idea for the analysis is that, when Eve changes Y_1, by the property of the non-malleable condenser, even conditioned on Z', Z has min-entropy $O(s)$. Thus when Y_2 is another random string independent of Y_1 and X, no matter what Y_2' is, as long as Y_2' is a function of Y_2, we can output $O(s)$ bits in $\bar{T}_1$, and $\bar{T}_1$ is independent of T_1'. Therefore the probability that Eve can come up with the correct $\bar{T}_1$ is small. However, there are several subtle problems here. First, T_1' may leak information about R or R', thus the MAC key may not be uniform. We can solve this problem by having the size of T_1' be small compared to R, so that R still has a lot of entropy left and thus we can use the leakage-resilient MAC. This ensures that when Eve does not change Y_1, the MAC can still authenticate W' to Alice. Second, Y_1', Y_2' can depend on both Y_1 and Y_2. We solve this problem by using the same idea we used before. We first fix Y_1 and Y_1' and by the property of the non-malleable condenser, for most choices of Y_1, no matter what Y_1' is, as long as $Y_1' \neq Y_1$, even conditioned on Z', Z has min-entropy $O(s)$. Now after this fixing Y_2' is indeed a deterministic function of Y_2. However, another problem arises. The problem is that fixing Y_1' may cause Y_2 to lose entropy (since Y_1' may depend on Y_2), and the tag T_2' may leak information about Z. Again, the solution is that we can make the size of Y_2 a constant times bigger than the size

of Y_1, so that fixing Y_1' doesn't cause Y_2 to lose much entropy. We can also limit the size of T_2' to be a constant times smaller than the conditional min-entropy of $Z|Z'$, so that even after fixing T_2', $Z|Z'$ still has min-entropy $O(s)$. Finally we need to use a strong two-source extractor to compute $\bar{T}_1$ and T_1'. We use the two-source extractor in [Raz05], which works as long as the seed Y_2 has entropy rate $> 1/2$. This gives our new protocol.

Organization. The rest of the paper is organized as follows. We give some preliminaries in Section 3. In Section 4 we show the existence of design extractors by the probabilistic method. In Section 5 we give explicit constructions of design extractors. In Section 6 we show how we can improve the output length of known non-malleable extractors. In Section 7 we introduce non-malleable condensers and show the applications in constructing non-malleable extractors and giving optimal privacy amplification protocols. Finally in Section 8 we conclude with some open problems.

3 Preliminaries

We often use capital letters for random variables and corresponding small letters for their instantiations. Let $|S|$ denote the cardinality of the set S. Let $\mathbb{Z}_r$ denote the cyclic group $\mathbb{Z}/(r\mathbb{Z})$, and let $\mathbb{F}_q$ denote the finite field of size q. All logarithms are to the base 2 unless otherwise stated.

3.1 Probability distributions

Definition 3.1 (statistical distance)**.** Let W and Z be two distributions on a set S. Their *statistical distance* (variation distance) is

$$\Delta(W,Z) \stackrel{def}{=} \max_{T \subseteq S}(|W(T) - Z(T)|) = \frac{1}{2}\sum_{s \in S}|W(s) - Z(s)|.$$

We say W is ε-close to Z, denoted $W \approx_\varepsilon Z$, if $\Delta(W,Z) \leq \varepsilon$. For a distribution D on a set S and a function $h : S \to T$, let $h(D)$ denote the distribution on T induced by choosing x according to D and outputting $h(x)$. We often view a distribution as a function whose value at a sample point is the probability of that sample point. Thus $\|W - Z\|_{\ell^1}$ denotes the ℓ_1 norm of the difference of the distributions specified by the random variables W and Z, which equals $2\Delta(W,Z)$.

Lemma 3.2 ([MW97])**.** *Let X and Y be random variables and let $\mathcal{Y}$ denote the range of Y. Then for all $\epsilon > 0$*

$$\Pr_Y\left[H_\infty(X|Y=y) \geq H_\infty(X) - \log|\mathcal{Y}| - \log\left(\frac{1}{\epsilon}\right)\right] \geq 1 - \epsilon$$

Definition 3.3. 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,

$$|(\mathsf{Ext}(X,R),R) - (U_m,R)| < \epsilon,$$

where R is the uniform distribution on d bits independent of X, and U_m is the uniform distribution on m bits independent of R.

Definition 3.4. A function $\mathsf{TExt} : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \to \{0,1\}^m$ is a *strong two source extractor* for min-entropy k_1, k_2 and error ϵ if for every independent (n_1, k_1) source X and (n_2, k_2) source Y,

$$|(\mathsf{TExt}(X,Y), X) - (U_m, X)| < \epsilon$$

and

$$|(\mathsf{TExt}(X,Y), Y) - (U_m, Y)| < \epsilon,$$

where U_m is the uniform distribution on m bits independent of (X, Y).

3.2 Average conditional min-entropy

Dodis and Wichs originally defined non-malleable extractors with respect to average conditional min-entropy, a notion defined by Dodis, Ostrovsky, Reyzin, and Smith [DORS08].

Definition 3.5. The *average conditional min-entropy* is defined as

$$\widetilde{H}_\infty(X|W) = -\log\left(\mathrm{E}_{w \leftarrow W}\left[\max_x \Pr[X = x|W = w]\right]\right) = -\log\left(\mathrm{E}_{w \leftarrow W}\left[2^{-H_\infty(X|W=w)}\right]\right).$$

Average conditional min-entropy tends to be useful for cryptographic applications. By taking W to be the empty string, we see that average conditional min-entropy is at least as strong as min-entropy. In fact, the two are essentially equivalent, up to a small loss in parameters.

We have the following lemmas.

Lemma 3.6 ([DORS08])**.** *For any $s > 0$, $\Pr_{w \leftarrow W}[H_\infty(X|W = w) \geq \widetilde{H}_\infty(X|W) - s] \geq 1 - 2^{-s}$.*

Lemma 3.7 ([DORS08])**.** *If a random variable B has at most 2^ℓ possible values, then $\widetilde{H}_\infty(A|B) \geq H_\infty(A) - \ell$.*

Corollary 3.8. *A (k, ε)-average-case non-malleable extractor is a (k, ε)-worst-case non-malleable extractor. For any $s > 0$, a (k, ε)-worst-case non-malleable extractor is a $(k + s, \varepsilon + 2^{-s})$-average-case non-malleable extractor.*

Throughout the rest of our paper, when we say non-malleable extractor, we refer to the worst-case non-malleable extractor of Definition 1.4.

3.3 Previous Work that We Use

For a strong seeded extractor with optimal parameters, we use the following extractor constructed in [GUV09].

Theorem 3.9 ([GUV09])**.** *For every constant $\alpha > 0$, and all positive integers n, k and any $\epsilon > 0$, there is an explicit construction of a strong (k, ϵ)-extractor $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ with $d = O(\log n + \log(1/\epsilon))$ and $m \geq (1 - \alpha)k$.*

We need the following two source extractor from [Raz05].

Theorem 3.10 ([Raz05])**.** *For any n_1, n_2, k_1, k_2, m and any $0 < \delta < 1/2$ with*

- $n_1 \geq 6 \log n_1 + 2 \log n_2$

- $k_1 \geq (0.5+\delta)n_1 + 3\log n_1 + \log n_2$
- $k_2 \geq 5\log(n_1 - k_1)$
- $m \leq \delta \min[n_1/8, k_2/40] - 1$

There is a polynomial time computable strong 2-source extractor $\mathsf{Raz} : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \to \{0,1\}^m$ *for min-entropy* k_1, k_2 *with error* $2^{-1.5m}$.

4 The Existence of Design Extractors

Here we show that design extractors with good parameters exist. Recall the definition of a design extractor.

Definition 4.1. An $(N, M, K, D, \alpha, \epsilon)$ design extractor is a bipartite graph with left hand side $[N]$, right hand side $[M]$, left degree D such that the following properties hold.

- (extractor property) For any subset $S \subseteq [M]$, let $\rho_S = |S|/M$. For any vertex $v \in [N]$, let $\rho_v = |\Gamma(v) \cap S|/D$. Let $Bad_S = \{v \in [N] : |\rho_v - \rho_S| > \epsilon\}$, then $|Bad_S| \leq K$.
- (design property) For any two different vertices $u, v \in [N]$, $|\Gamma(u) \cap \Gamma(v)| \leq \alpha D$.

Theorem 4.2. *There exist design extractors such that* $C\max\{\log(N/K)/\epsilon^2, \log N/\alpha\} \leq D \leq \alpha M/2$ *and* $M < \beta\epsilon^2 DK$ *for some constants* $C > 1, \beta < 1$.

Proof. We show that a random bipartite graph is a design extractor with the above parameters with high probability. Specifically, For each vertex $v \in [N]$, we take its D neighbors independently and uniformly at random from $[M]$. We then bound the probability that the graph does not satisfy either property of a design extractor.

The first property is essentially the extractor property. Thus we have that for any vertex $v \in [N]$ and any subset $S \subseteq [M]$, the expected value of $|\Gamma(v) \cap S|$ is $\rho_S D$. Therefore we have

$$\Pr[|\rho_v - \rho_S| > \epsilon] < 2^{-\Omega(\epsilon^2 D)}$$

by a chernoff bound.

If $|Bad_S| > K$, then there exists such a set with size K, and this happens with probability at most $(2^{-\Omega(\epsilon^2 D)})^K = 2^{-\Omega(\epsilon^2 DK)}$ since the vertices choose their neighbors independently. Thus by the union bound, the probability that the first property is not satisfied is at most

$$\binom{N}{K} 2^M 2^{-\Omega(\epsilon^2 DK)}.$$

Now consider the second property. For any two different vertices u and v in $[N]$, since each of them chooses its neighbors independently, the expected value of $|\Gamma(u) \cap \Gamma(v)|$ is $D/M \cdot D \leq \alpha D/2$ since $D \leq \alpha M/2$. Therefore we have

$$\Pr[|\Gamma(u) \cap \Gamma(v)| > \alpha D] < 2^{-\Omega(\alpha D)}$$

by a chernoff bound.

Thus by the union bound, the probability that the second property is not satisfied is at most

$$\binom{N}{2}2^{-\Omega(\alpha D)}.$$

Therefore the probability that a random graph is not a design extractor is at most

$$\binom{N}{K}2^{M}2^{-\Omega(\epsilon^2 DK)} + \binom{N}{2}2^{-\Omega(\alpha D)}.$$

It is easy to check this probability is exponentially small when we choose the parameters as in the theorem statement. ■

5 Explicit Constructions of Design Extractors

We first show that a design extractor can be constructed from an extractor with appropriate parameters.

Lemma 5.1. *For integers M, D and $0 < \epsilon, \alpha < 1$ such that $D \leq \alpha M/2$, an $(N, M, K, D, \alpha, \epsilon)$ design extractor can be constructed from an (N', M, K, D, ϵ') extractor in time* $\mathrm{poly}(N')$ *where $N = N'/K$ and $\epsilon' = \min\{\epsilon, \alpha/2\}$.*

Proof. Suppose that there exists an (N', M, K, D, ϵ') extractor. Note that $\epsilon \geq \epsilon'$, thus the graph already satisfies the first property of the design extractor. Next, for any particular vertex $u \in [N']$, let $S_u = \Gamma(u)$, $\rho_{S_u} = |S_u|/M$. Now for any different vertex $v \in [N']$, let $\rho_v = |\Gamma(v) \cap S_u|/D$ and let $Bad_{S_u} = \{v \in [N'] : |\rho_v - \rho_{S_u}| > \epsilon'\}$. Note that $\rho_{S_u} + \epsilon' \leq \alpha$. Thus any vertex $v \in [N']$ such that $|\Gamma(v) \cap \Gamma(u)| > \alpha D$ is in Bad_{S_u}. By the extractor property $|Bad_{S_u}| \leq K$. Thus we conclude that for any vertex $u \in [N']$, there are at most $K - 1$ different vertices v in $[N']$ such that $|\Gamma(u) \cap \Gamma(v)| > \alpha D$.

Now we can construct the design extractor by picking vertices in $[N]$ from $[N']$ greedily. Specifically, we first pick any vertex $u \in [N']$ and put it in $[N]$. Next, we delete all different vertices v in $[N']$ such that $|\Gamma(v) \cap \Gamma(u)| > \alpha D$. We then pick any vertex from the remaining vertices, and keep on doing this. Since each step we delete at most $K - 1$ vertices, we have at least $N = N'/K$ vertices left. Finally note that deleting vertices from $[N']$ does not change the extractor property. Thus we get an $(N, M, K, D, \alpha, \epsilon)$ design extractor. □

The above lemma shows that a design extractor can be constructed from an extractor with appropriate parameters, but the construction is only weakly efficient in the sense that the running time is polynomial in the number of vertices. Ideally we would want a construction that is strongly efficient in the sense that the running time is polylogarithmic in the number of vertices. We next show that we can achieve this by using a special kind of extractors–Trevisan's extractor [Tre01].

Construction 5.2 (Trevisan's extractor)**.** [Tre01] $\mathsf{Trev} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$.

- Let $C : \{0,1\}^n \to \{0,1\}^{\bar{n}}$ be a binary code.
- Let $l = \log \bar{n}$ and $S = \{S_1, \cdots, S_m\}$ be a design such that $\forall i, S_i \subseteq [d]$ and $|S_i| = l$.

For a seed $r \in \{0,1\}^d$, let $j_i \in [\bar{n}]$ be the number whose binary expression is $r|_{S_i}$, the i' th bit of Trevisan's extractor $\mathsf{Trev}(x, r)$ is defined as the j_i'th bit of $C(x)$.

Based on Trevisan's extractor $\mathsf{Trev} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$, construct a bipartite graph DExt with left hand side $[N]$, right hand side $[M]$ and left degree D, where $N = 2^n, M = 2^{m+d}, D = 2^d$ as follows. For every $v \in [N]$ let x_v be its binary representation. For every $r \in \{0,1\}^d$ connect v to the vertex in $[M]$ whose binary representation is $\mathsf{Trev}(x_v, r) \circ r$.

We now have the following lemma.

Lemma 5.3. *Assume* Trev *is an* (n, k, ϵ) *extractor and the binary code in Trevisan's extractor has relative distance* $1 - \alpha$*. Then* DExt *is an* $(N, M, K, D, \alpha, \epsilon)$ *design extractor with* $K = 2^k$*.*

Proof. Note that the right hand side of the graph, which corresponds to the output of the extractor, is essentially $\mathsf{Trev}(X, R) \circ R$. Since Trev is a strong extractor, after concatenation with the seed, it is still an (n, k, ϵ) extractor. Thus DExt satisfies the extractor property. Next, consider two different vertices $u, v \in [N]$ and their neighbors $\Gamma(u), \Gamma(v)$. Since a neighbor with edge r is represented by $\mathsf{Trev}(x_v, r) \circ r$, if a neighbor of u and a neighbor of v are the same, they must be with the same edge r. Now consider the first bit of $\mathsf{Trev}(x_u, r)$ and $\mathsf{Trev}(x_v, r)$. These are basically the j_1'th bit in $C(x_u)$ and $C(x_v)$, where j_1 is the number whose binary expression is $r|_{S_1}$. Note S_1 is a subset of $[d]$. Thus as r is uniformly distributed over $\{0,1\}^d$, j_1 is also uniformly distributed over $\{0,1\}^l$. Therefore by the distance property of the binary code, the first bits of $\mathsf{Trev}(x_u, r)$ and $\mathsf{Trev}(x_v, r)$ differ in at least $(1 - \alpha)D$ positions. Thus $|\Gamma(u) \cap \Gamma(v)| \leq \alpha D$. □

The above lemma shows that in order to get a design extractor, all we need is to take a binary code with large distance in Trevisan's extractor. However, a binary code cannot have relative distance greater than $1/2$, so the parameter α cannot be less than $1/2$. Sometimes we would want α to be much smaller, and in this case we need to take a code over a larger alphabet. Specifically, we are going to use the following code that have good combinatorial list decoding property.

We first have the following theorem.

Theorem 5.4. *[GS00] For a* q*-ary code of block length* n *and distance* $d = (1 - 1/q)(1 - \delta)n$*, for any received word, the number of codewords differing from the received word in at most* e *places, where* $qe/(q-1) = (1-\gamma)n$*, is at most* $\frac{1-\delta}{\gamma^2 - \delta}$*, provided* $\gamma > \sqrt{\delta}$*.*

Next we use one of the concatenated codes constructed in [GS00].

Theorem 5.5. *[GS00] For every* $k, \delta > 0$*, there is an explicitly specified* q*-ary code* C*, obtained by concatenating a Reed-Solomon code with a Hadamard code, which has rate* k*, relative distance* $d/n = (1 - 1/q)(1 - \delta)$*, and block length* $n = O\left(\frac{k^2}{\delta^2 \log^2(1/\delta)}\right)^2$*.*

Trevisan's extractor uses a weak design, for which we have the following definition and theorem.

Definition 5.6. [RRV02] A family of sets $S_1, \cdots, S_m \subset [d]$ is a weak (ℓ, ρ)-design if

1. For all i, $|S_i| = \ell$.
2. For all i, $\sum_{j<i} 2^{|S_i \cap S_j|} \leq \rho \cdot (m - 1)$.

Theorem 5.7. *[RRV02] For every* $\ell, m \in \mathbb{N}$ *and* $\rho > 1$*, there exists a weak* (ℓ, ρ)*-design* $S_1, \cdots, S_m \subset [d]$ *with*

$$d = \left\lceil \frac{\ell}{\ln \rho} \cdot \ell \right\rceil.$$

Moreover, such a family can be found in time $\mathrm{poly}(m, d)$*.*

We now define the following generalized version of Trevisan's extractor, over some alphabet $[q]$.

Construction 5.8 (Generalized Trevisan's extractor)**.** $\mathsf{GTrev} : [q]^n \times \{0,1\}^d \to [q]^m$.

- Let $C : [q]^n \to [q]^{\bar{n}}$ be the code in Theorem 5.5.
- Let $l = \log \bar{n}$ and $S = \{S_1, \cdots, S_m\} \subset [d]$ be an (l, ρ) design for some parameter $\rho > 1$.

For a seed $r \in \{0,1\}^d$, let $j_i \in [\bar{n}]$ be the number whose binary expression is $r|_{S_i}$, the i' th symbol of the generalized Trevisan's extractor $\mathsf{GTrev}(x, r)$ is defined as the j_i'th symbol of $C(x)$.

Again, we construct a bipartite graph DExt based on the above generalized Trevisan's extractor. The graph has left hand side $[N]$, right hand side $[M]$ and left degree D, where $N = q^n, M = q^m 2^d$ and $D = 2^d$. The graph is defined as follows. For every $v \in [N]$ let x_v be its q-ary representation. For every $r \in \{0,1\}^d$ connect v to the vertex in $[M]$ which is represented by $\mathsf{Trev}(x_v, r) \circ r$.

We now analyze the above construction.

Theorem 5.9. *For any integers N, K such that $K = N^{\Omega(1)}$ and any constants $\epsilon, \alpha > 0$, let $q = 2/\alpha$, $n = \log_q N$, $k = \log_q K$ and $m = \sqrt{k}$. Let $\mathsf{GTrev} : [q]^n \times \{0,1\}^d \to [q]^m$ be the generalized Trevisan's extractor as described above, with parameters q, n, m and $\rho = (k - 2\log_q(m/\epsilon) - O(1))/m$. Then DExt is an $(N, M, K, D, \alpha, \epsilon)$ design extractor with $D = \text{polylog}(N)$ and $M = q^{\sqrt{k}}\text{polylog}(N)$.*

Proof. We first show that GTrev is indeed a strong extractor. Let U_m denote the uniform distribution over $[q]^m$. Without loss of generality assume that for some test T (i.e., a subset of $[D] \times [M]$),

$$\Pr[T(R, \mathsf{GTrev}(X, R)) = 1] - \Pr[T(R, U_m) = 1] \geq \epsilon.$$

Define the hybrid distributions $D_0, ..., D_m$ where D_i is the concatenation of R, the first i symbols from $\mathsf{GTrev}(X, R)$ and the last $m - i$ symbols from U_m. Thus we have

$$\Pr[T(D_m) = 1] - \Pr[T(D_0) = 1] \geq \epsilon.$$

Therefore there must be an i, $1 \leq i \leq m$, such that

$$\Pr[T(D_i) = 1] - \Pr[T(D_{i-1}) = 1] \geq \frac{\epsilon}{m}.$$

Let D'_i be the distribution obtained by randomly choosing a different symbol of the $(d+i)$'th symbol in D_i (the i'th symbol in $\mathsf{GTrev}(X, R)$), and $p_i = \Pr[T(D_i) = 1]$, $s_i = \Pr[T(D'_i) = 1]$, $p_{i-1} = \Pr[T(D_{i-1}) = 1]$. Then

$$p_{i-1} = \frac{1}{q} \cdot p_i + (1 - \frac{1}{q}) \cdot s_i.$$

Thus

$$p_i - s_i = \frac{q}{q-1}(p_i - p_{i-1}).$$

Let the $(d+i)$'th symbol in D_{i-1}(the first symbol chosen from U_m) be u_1. Now let T_2 be a randomized circuit that first runs T on D_{i-1}, and then predicts the i'th symbol in $\mathsf{GTrev}(X, R)$, y_i, as follows: if $T(D_{i-1}) = 1$, predict $y_i = u_1$; otherwise choose a symbol y uniformly at random from $[q]/\{u_1\}$ and predict $y_i = y$. We have

$$\begin{aligned}
\Pr[\mathit{correct}] &= \Pr[\mathit{correct}|y_i = u_1]\Pr[y_i = u_1] + \Pr[\mathit{correct}|y_i \neq u_1]\Pr[y_i \neq u_1] \\
&= p_i \cdot \frac{1}{q} + \frac{1}{q-1}(1-s_i)\cdot(1-\frac{1}{q}) \\
&= \frac{1}{q} + \frac{1}{q}\cdot(p_i - s_i) \\
&= \frac{1}{q} + \frac{1}{q-1}\cdot(p_i - p_{i-1}) \\
&\geq \frac{1}{q} + \frac{1}{q-1}\cdot\frac{\epsilon}{m}
\end{aligned}$$

Thus, we have a randomized circuit that predicts the i'th symbol of $\mathsf{GTrev}(X,R)$ from the first $i-1$ symbols with advantage $\epsilon/((q-1)m)$. There is a fixing of the coin tosses of the circuit that preserves the advantage. Also, note that in D_{i-1} the last $m-i+1$ symbols are uniform and independent of all the other symbols. Hence there exists a fixing of these symbols that preserves the advantage of at least $\epsilon/((q-1)m)$. Furthermore, we can split $R \in \{0,1\}^d$ into those bits in locations indexed by S_i and the rest. Again, there is a way to fix the rest bits and still preserve the advantage. Now we have a circuit T_3 with all the fixings hardwired, such that T_3 predicts the i'th symbol of $\mathsf{GTrev}(X,R)$ from the first $i-1$ symbols with probability at least $\frac{1}{q} + \frac{1}{q-1}\cdot\frac{\epsilon}{m}$.

Note that the only random bits left in R are the bits indexed by S_i. If we know the dependence of the first $i-1$ symbols of $\mathsf{GTrev}(X,R)$ on these bits, then by trying all possibilities of the bits in S_i we get a string w that agrees with $C(x)$ in at least $(\frac{1}{q} + \frac{\epsilon}{(q-1)m})\bar{n}$ coordinates. Thus, the codeword differs with w in at most $e = (1 - \frac{1}{q} - \frac{\epsilon}{(q-1)m})\bar{n}$ positions. Thus we can use Theorem 5.4 to bound the number of such codewords. Specifically, γ there is $\frac{q}{(q-1)^2}\cdot\frac{\epsilon}{m}$. We choose $\delta = \frac{q^2}{2(q-1)^4}\cdot\frac{\epsilon^2}{m^2}$ so that the conditions in Theorem 5.4 are met. Thus we have that the number of codewords is at most

$$\frac{1-\delta}{\gamma^2-\delta} < \frac{1}{\gamma^2-\delta} = \frac{2(q-1)^4}{q^2}\cdot\frac{m^2}{\epsilon^2}.$$

Since we don't know the dependence, we try all possibilities. Thus, the number of codewords is at most

$$q^{\sum_{j<i} 2^{|S_i \cap S_j|}} \cdot \frac{2(q-1)^4}{q^2}\cdot\frac{m^2}{\epsilon^2} \leq q^{\rho m}\cdot\frac{2(q-1)^4}{q^2}\cdot\frac{m^2}{\epsilon^2}.$$

Therefore we can take $\rho = (k - 2\log_q(m/\epsilon) - O(1))/m = k^{\Omega(1)}$ and the above number is at most $K = q^k$. Note $\bar{n} = O(\frac{n^2}{\delta^2\log^2(1/\delta)})^2 = O_q(\frac{n^2m^4}{\epsilon^4\log^2(m/\epsilon)})$. Thus

$$d = \left\lceil \frac{\log \bar{n}}{\ln \rho} \right\rceil \cdot \log \bar{n} = O\left(\frac{\log^2 n}{\log k}\right) = O(\log n).$$

Thus we have shown that GTrev is a strong extractor. Therefore DExt is an extractor with $D = \mathrm{polylog}(N)$ and $M = q^{\sqrt{k}}\mathrm{polylog}(N)$. Next note that $q = 2/\alpha$. Thus by Theorem 5.5 the relative distance of the code is at least $(1-1/q)(1-\delta) > 1-\alpha$. Now by the same argument as in the proof of Lemma 5.3 we have that DExt is an $(N, M, K, D, \alpha, \epsilon)$-design extractor. □

In the above construction we only have that $D = \text{polylog}(N) = \text{polylog}(M)$. Sometimes in our applications we will want D to be large, e.g., $D = \Omega(M)$. We have the following theorem.

Theorem 5.10. *For any integers N, K, M such that $K = N^{\Omega(1)}, M \geq \text{polylog}(N)$ and any constants $\epsilon, \alpha, \beta > 0$. There exists an $(N, M, K, D, \alpha, \epsilon)$ design extractor such that $D = \beta M$. Moreover the neighbors of each vertex in $[N]$ can be computed in time $\text{poly}(\log N, M)$.*

Proof. We first use Theorem 5.9 to construct an $(N, M', K, D', \alpha, \epsilon)$ design extractor such that $D' = \text{polylog}(N)$ and $D' = \beta M'$. This simply corresponds to the case where we only output some constant $c = \log(1/\beta)$ number of bits in GTrev. Next, we add the same amount of independent random bits to both the seed and the output of GTrev until the output has $\log M$ bits. Now we have $D = \beta M$. It is easy to check that the resulting graph is an $(N, M, K, D, \alpha, \epsilon)$ design extractor. ■

6 Improving the Output Length of Non-Malleable Extractors

In this section we show how we can use design extractors to improve the output length of non-malleable extractors. Assume that we have an (n, k)-source with $k = (1/2 + \delta)n$ for an arbitrary constant $0 < \delta < 1$. We first need an $(N, M, K, D, \alpha, \epsilon)$ design extractor such that $M = n$. We are going to associate the right hand side of the design extractor with the bits of the weak random source.

We need the following definition of an average sampler.

Definition 6.1. [Vad04] A function $\mathsf{Samp} : \{0,1\}^r \to [n]^t$ is a (μ, ϵ, γ) averaging sampler if for every function $f : [n] \to [0,1]$ with average value $\frac{1}{n}\sum_i f(i) \geq \mu$, it holds that [1]

$$\Pr_{(i_1,\cdots,i_t)\leftarrow\mathsf{Samp}(U_r)}\left[\frac{1}{t}\sum_{j=1}^{t} f(i_j) < \mu - \epsilon\right] \leq \gamma.$$

Samp has *distinct samples* if for every $x \in \{0,1\}^r$, the samples produced by $\mathsf{Samp}(x)$ are all distinct.

It is shown in [Zuc97] that randomness extractors give averaging samplers (in fact, these two objects are equivalent). We have the following theorem.

Theorem 6.2. *[Zuc97] An (N, M, K, D, ϵ) extractor is a (μ, ϵ, γ) averaging sampler for any $\mu > 0$, $t = D$ and $\gamma = K/N$.*

To ensure that the sampler has distinct samples, we let the output of the extractor be the concatenation of a strong extractor's output and the seed. Thus different seeds correspond to different vertices on the right.

The idea is that, we will use our design extractor as the averaging sampler to sample bits from a weak random source X. Specifically, this is done by picking a vertex uniformly from the left hand side, and then use the concatenation of its D neighbors (D bits) as the output of the sampler. Note that this only requires $d_1 = \log N$ bits. We now have the following two lemmas about the output of the sampler.

[1] In general average samplers consider two sided errors, however in this paper it is enough to consider one-sided error, as in [Vad04]. Moreover, two sided error are actually implied by one sided error by considering $1 - f$.

Let X be a weak random source with min-entropy δn for some $\delta > 0$. Let $S \in [n]^D$ be the output of the sampler, we write X_i for the i'th bit of X and X_S for the bits of X that are indexed by S. Some part of our analysis will follow that of [Vad04].

For every $x \in \mathsf{Supp}(X)$, define $p_i(x) = \Pr[X_i = x_i | X_1 = x_1, \cdots, X_{i-1} = x_{i-1}]$ and $h_i(x) = \log(1/p_i(x))$. Intuitively, if read in a stream from x_1 to x_n, $h_i(x)$ would be the min-entropy that is contributed by the bit x_i. Since X has min-entropy δn, we have $\Pr[X = x] = \Pi_i p_i(x) = 2^{-\sum_i h_i(x)} \leq 2^{-\delta n}$. Thus $\sum_i h_i(x) \geq \delta n$, which means that the average of $h_i(x)$ is at least δ for every $x \in \mathsf{Supp}(X)$. Therefore with high probability the sampler will output a sequence of bits with entropy rate roughly δ. However, the sampler is only guaranteed to work for functions with output range $[0, 1]$, while $h_i(x)$ could be greater than 1. Thus we will set a threshold and truncate large values. Specifically, we choose a parameter $\tau < \delta/3$ and let $h'_i(x) = min\{h_i(x), \log(1/\tau)\}$.

We need to argue that this truncation does not cost us much entropy. To this end, call x *well-spread* if $\sum_i h'_i(x) \geq (\delta - 2\tau)n$. We have the following lemma.

Lemma 6.3. *[Vad04]* $\Pr[X \text{ is not well-spread}] \leq 2^{-\Omega(\tau n)}$.

Next, choose $\mu = (\delta - 2\tau)/\log(1/\tau)$ and $\epsilon = \tau/\log(1/\tau)$ in the sampler. We will call a sequence $s = (i_1, \cdots, i_D) \in [n]^D$ *good* for x if

$$\frac{1}{D}\sum_{j=1}^{D} h'_{i_j}(x) \geq \delta - 3\tau.$$

Otherwise we will call s *bad* for x. Let $b(s) = \Pr[s \text{ is bad for } X]$. We have

Lemma 6.4. *[Vad04]* $E[b(S)] \leq \gamma + 2^{-\Omega(\tau n)}$.

Next, we show that the bits indexed by good sequences have high min-entropy. The proof is essentially the same as in [Vad04], but we include it both for completeness and our next lemma.

Lemma 6.5. *[Vad04] For every s, the random variable X_s is $b(s)$-close to a $(\delta - 3\tau)D$ -source.*

Proof. Fix some $s = (i_1, \cdots, i_D)$. Now think of X as being generated by the following process. We have n independent random variables $F_1, \cdots, F_n$ where F_i is distributed (arbitrarily) over functions from $\{0,1\}^{i-1}$ to $\{0,1\}$, and X is deterministically generated by setting $X_i = F_i(X_1, X_2, \cdots, X_{i-1})$.

We will now fix F_i for every $i \notin s$. For any fixing $\bar{f} = (f_i)_{i \notin s}$, let $X^{\bar{f}}$ denote X conditioned on $\bar{f}$. Note that for any string $z \in \{0,1\}^D$ in the support of $X_s^{\bar{f}}$, there is a unique $x \in \{0,1\}^n$ in the support of $X^{\bar{f}}$ such that $x_s = z$ (which is the string obtained by setting $x_i = z_i$ for all $i \in s$ and $x_i = f_i(x_1, \cdots, x_{i-1})$ for all $i \notin s$.). Let this x be $x^{\bar{f}}(z)$. We have

$$\Pr[X_s^{\bar{f}} = z] = \Pr[X^{\bar{f}} = x^{\bar{f}}(z)] = \Pi_{i=1}^{n}[X_i^{\bar{f}} = x^{\bar{f}}(z)_i | X_1^{\bar{f}} = x^{\bar{f}}(z)_1, \cdots, X_{i-1}^{\bar{f}} = x^{\bar{f}}(z)_{i-1}].$$

Note that when $i \notin s$, the conditional probability in the i'th factor is 1, and when $i \in s$, the conditional probability equals $p_i(x^{\bar{f}}(z))$. Therefore if s is good for $x^{\bar{f}}(z)$ and consists of distinct samples, we have

$$\Pr[X_s^{\bar{f}} = z] = \Pi_{i \in s} p_i(x^{\bar{f}}(z)) \leq \Pi_{i \in s} 2^{-h'_i(x^{\bar{f}}(z))} \leq 2^{-(\delta - 3\tau)D}.$$

Let $b(s,\bar{f})$ denote the probability that s is bad for $X^{\bar{f}}$. Then $X_s^{\bar{f}}$ is $b(s,\bar{f})$-close to some $(\delta-3\tau)D$-source $Z^{\bar{f}}$. Now consider the random variable $\bar{F}=(F_i)_{i\notin s}$. Note that $X_s=X_s^{\bar{F}}$ and $Z^{\bar{F}}$ is a convex combination of $(\delta-3\tau)D$-sources and thus is also a $(\delta-3\tau)D$-source. Moreover the statistical distance between X_s and $Z^{\bar{F}}$ is at most $E(b(s,\bar{F}))=b(s)$. □

We now have the following lemma.

Lemma 6.6. *Assume the averaging sampler is an $(N,M,K,D,\alpha,\epsilon)$ design extractor. Then for every two different s_1,s_2, the random variable (X_{s_1},X_{s_2}) is $b(s_1)+b(s_2)$-close to a $(2\delta-6\tau-\alpha\log(1/\tau))D$ source.*

Proof. Similar as in the proof of Lemma 6.5, fix some $s_1=(l_1,\cdots,l_D)$ and $s_2=(j_1,\cdots,j_D)$. Let $s=s_1\cup s_2$. Note that $(X_{s_1},X_{s_2})=X_s$.

Again we fix F_i for every $i\notin s$. For any fixing $\bar{f}=(f_i)_{i\notin s}$, let $X^{\bar{f}}$ denote X conditioned on $\bar{f}$. Note that for any string $z\in\{0,1\}^D$ in the support of $X_s^{\bar{f}}$, there is a unique $x\in\{0,1\}^n$ in the support of $X^{\bar{f}}$ such that $x_s=z$ (which is the string obtained by setting $x_i=z_i$ for all $i\in s$ and $x_i=f_i(x_1,\cdots,x_{i-1})$ for all $i\notin s$.). Let this x be $x^{\bar{f}}(z)$. We have

$$\Pr[X_s^{\bar{f}}=z]=\Pr[X^{\bar{f}}=x^{\bar{f}}(z)]=\Pi_{i=1}^n[X_i^{\bar{f}}=x^{\bar{f}}(z)_i|X_1^{\bar{f}}=x^{\bar{f}}(z)_1,\cdots,X_{i-1}^{\bar{f}}=x^{\bar{f}}(z)_{i-1}].$$

Note that when $i\notin s$, the conditional probability in the i'th factor is 1, and when $i\in s$, the conditional probability equals $p_i(x^{\bar{f}}(z))$. Therefore we have

$$\Pr[X_s^{\bar{f}}=z]=\Pi_{i\in s}p_i(x^{\bar{f}}(z))\le\Pi_{i\in s}2^{-h_i'(x^{\bar{f}}(z))}.$$

Note that by the property of the design extractor, the intersection of s_1 and s_2 has at most αD elements. Also note that $h_i'(x^{\bar{f}}(z))\le\log(1/\tau)$. Thus we have

$$\Pi_{i\in s}2^{-h_i'(x^{\bar{f}}(z))}\le\Pi_{i\in s_1}2^{-h_i'(x^{\bar{f}}(z))}\cdot\Pi_{i\in s_2}2^{-h_i'(x^{\bar{f}}(z))}\cdot(2^{\log(1/\tau)})^{\alpha D}.$$

Thus if both s_1 and s_2 are good for $x^{\bar{f}}(z)$, we have

$$\Pr[X_s^{\bar{f}}=z]\le 2^{-(\delta-3\tau)D}\cdot 2^{-(\delta-3\tau)D}\cdot(2^{\log(1/\tau)})^{\alpha D}=2^{-(2\delta-6\tau-\alpha\log(1/\tau))D}.$$

Let $b(s,\bar{f})$ denote the probability that either s_1 or s_2 is bad for $X^{\bar{f}}$. Note that $b(s,\bar{f})\le b(s_1,\bar{f})+b(s_2,\bar{f})$ by the union bound. Then $X_s^{\bar{f}}$ is $b(s,\bar{f})$-close to some $(2\delta-6\tau-\alpha\log(1/\tau))D$-source $Z^{\bar{f}}$. Now consider the random variable $\bar{F}=(F_i)_{i\notin s}$. Note that $X_s=X_s^{\bar{F}}$ and $Z^{\bar{F}}$ is a convex combination of $(2\delta-6\tau-\alpha\log(1/\tau))D$-sources and thus is also a $(2\delta-6\tau-\alpha\log(1/\tau))D$-source. Moreover the statistical distance between X_s and $Z^{\bar{F}}$ is at most $E(b(s,\bar{F}))\le E(b(s_1,\bar{f}))+E(b(s_2,\bar{f}))=b(s_1)+b(s_2)$. □

We now consider the case where the weak random source X has min-entropy rate $1/2+\delta$ for some constant $\delta>0$. We choose the parameters such that $\tau=\delta/10$, $\alpha=\tau/\log(1/\tau)$ and $D=\Omega(\alpha M)=\Omega(n)$. We have the following lemma.

Lemma 6.7. *Assume X has min-entropy $k=(1/2+\delta)n$ for some constant $\delta>0$. Let Y be a uniform seed used by the sampler, and let $\mathcal{A}:\{0,1\}^r\to\{0,1\}^r$ be any deterministic function such that $\forall y,\mathcal{A}(y)\neq y$. Let $s=\mathsf{Samp}(y)$ and $A(s)=\mathsf{Samp}(\mathcal{A}(y))$ where Samp is the $(N,M,K,D,\alpha,\epsilon)$*

design extractor described above. Then for at least $1 - O\left(\sqrt{\frac{K^3}{N}}\right)$ *fraction of the choices of* s*, with probability at least* $1 - \sqrt[12]{\gamma} - 2^{-\Omega(\delta n)}$ *over the fixing of* $X_{A(s)}$*,* X_s *is* $\sqrt[6]{\gamma} + 2^{-\Omega(\delta n)}$*-close to a* $\delta D/2$ *source.*

Proof. First note that by Lemma 6.4, $E(b(s)) \leq \gamma + 2^{-\Omega(\tau n)} = \gamma + 2^{-\Omega(\delta n)}$. Thus by Markov's inequality we have

$$\Pr[b(s) > \sqrt[4]{\gamma} + 2^{-\Omega(\delta n)}] < \sqrt[4]{\gamma^3} + 2^{-\Omega(\delta n)}.$$

Now we will say s is good if $b(s) \leq \sqrt[4]{\gamma} + 2^{-\Omega(\delta n)}$, otherwise we say s is bad. Note the notion of "good" here is different from those in Lemma 6.5 and Lemma 6.6. Note that

$$2(1/2 + \delta) - 6\tau - \alpha \log(1/\tau) = 1 + 2\delta - 7\tau > 1 + \delta.$$

Thus for two different s_i, s_j, if both of them are good, then by Lemma 6.6 we have that (X_{s_i}, X_{s_j}) is $2\sqrt[4]{\gamma} + 2^{-\Omega(\delta n)}$-close to a $(1+\delta)D$ source. Note that X_{s_j} only has D bits, therefore by Lemma 3.2 we have

$$\Pr_{X_{s_j}}[X_{s_i}|_{X_{s_j}=x} \text{ is } 2\sqrt[4]{\gamma} + 2^{-\Omega(\delta n)} \text{ close to a } \delta D/2 \text{ source}] \geq 2^{-\Omega(\delta D)}.$$

Next, we consider those bad s. Fix a bad $\bar{s}$ and let $\bar{X}$ denote $X_{\bar{s}}$. Note that the random variable S is sampled from N possible sequences using $d_1 = \log N$ bits.

Note that $\bar{X}$ has only D bits and $D < \delta n/10$. Thus by Lemma 3.2 we have

$$\Pr_{\bar{X}}[X|_{\bar{X}=\bar{x}} \text{ has min-entropy } n/2] \geq 1 - 2^{-\Omega(\delta n)}.$$

When $X|_{\bar{X}=\bar{x}}$ has min-entropy $n/2$, again by Lemma 6.4 and Lemma 6.5 we have that $X_s|_{\bar{X}=\bar{x}}$ is $b(s)$-close to a $(1/2 - 3\tau)D$-source and

$$E[b(S)] \leq \gamma + 2^{-\Omega(\tau n)} = \gamma + 2^{-\Omega(\delta n)}.$$

Thus we have

$$E_{\bar{X},S}[b(s)] \leq \gamma + 2^{-\Omega(\delta n)} + 2^{-\Omega(\delta n)} = \gamma + 2^{-\Omega(\delta n)}.$$

Therefore

$$\Pr_{\bar{X},S}[b(s) > \sqrt[6]{\gamma} + 2^{-\Omega(\delta n)}] < \sqrt[5]{\gamma^6} + 2^{-\Omega(\delta n)}.$$

Thus

$$\Pr_{S}[\Pr_{\bar{X}}[b(s) > \sqrt[6]{\gamma} + 2^{-\Omega(\delta n)}] > \sqrt[12]{\gamma} + 2^{-\Omega(\delta n)}] < \sqrt[4]{\gamma^3} + 2^{-\Omega(\delta n)}.$$

In other words, for any bad $\bar{s}$, with probability at least $1 - \sqrt[4]{\gamma^3} - 2^{-\Omega(\delta n)}$ over the choices of $S = s$, with probability at least $1 - \sqrt[12]{\gamma} - 2^{-\Omega(\delta n)}$ over the choices of $\bar{X}$, $X_s|_{\bar{X}=\bar{x}}$ is $\sqrt[6]{\gamma} + 2^{-\Omega(\delta n)}$-close to a $(1/2 - 3\tau)$D source.

Note that $1/2 - 3\tau > \delta/2$. Since there are at most $(\sqrt[4]{\gamma^3} + 2^{-\Omega(\delta n)})N$ bad s, by a union bound they can ruin at most

$$(\sqrt[4]{\gamma^3} + 2^{-\Omega(\delta n)})N(\sqrt[4]{\gamma^3} + 2^{-\Omega(\delta n)}) = O(\gamma^{3/2}N) = O\left(\sqrt{\frac{K^3}{N}}\right)$$

fraction of S, as long as $\gamma \geq 2^{-\Omega(\delta n)}$. Thus when s does not belong to this bad fraction, no matter what $s' = A(s)$ is, we have that with probability at least $1 - \sqrt[12]{\gamma} - 2^{-\Omega(\delta n)}$ over the choices of $X_{s'}$, X_s is $\sqrt[6]{\gamma} + 2^{-\Omega(\delta n)}$-close to a $\delta D/2$ source. □

Note that $\gamma = K/N$. Thus we have the following corollary.

Corollary 6.8. *For any constant $\delta > 0$, there exists a constant $\beta > 0$ such that for any $\epsilon > 2^{-\beta n}$, there exists an explicit design extractor with $N = \mathrm{poly}(1/\epsilon), M = n, D = \Omega(n)$ and the following holds. Assume X has min-entropy $k = (1/2+\delta)n$. Let Y be a uniform seed used by the sampler, and let $\mathcal{A} : \{0,1\}^r \to \{0,1\}^r$ be any deterministic function such that $\forall y, \mathcal{A}(y) \neq y$. Let $s = \mathsf{Samp}(y)$ and $A(s) = \mathsf{Samp}(\mathcal{A}(y))$ where Samp is the $(N, M, K, D, \alpha, \epsilon)$ design extractor described above. With probability at least $1 - \epsilon$ over the choices of $S = s$, we have that with probability at least $1 - \epsilon$ over the choices of $X_{A(s)}$, X_s is ϵ-close to a $\delta D/2$ source.*

Note that here for a fixed source X the ϵ fraction of bad s are the same even for different functions $\mathcal{A}$. This is important for us.

Now we show how the above lemma can be used to construct a non-malleable extractor with large output length. First we need the following definition and theorem about non-malleable extractors with weak random seeds.

Definition 6.9. [DLWZ11] A function $\mathsf{nmExt} : [N] \times [D] \to [M]$ is a (k, k', ε)-non-malleable extractor if, for any source X with $H_\infty(X) \geq k$, any seed Y with $H_\infty(Y) \geq k'$, and any function $\mathcal{A} : [D] \to [D]$ such that $\mathcal{A}(y) \neq y$ for all y, the following holds:

$$(\mathsf{nmExt}(X, Y), \mathsf{nmExt}(X, \mathcal{A}(Y)), Y) \approx_\varepsilon (U_{[M]}, \mathsf{nmExt}(X, \mathcal{A}(Y)), Y).$$

A non-malleable extractor with small error will remain to be non-malleable even if the seed is somewhat weak random.

Lemma 6.10. *[DLWZ11] A (k, ϵ)-non-malleable extractor $\mathsf{nmExt} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is also a (k, k', ϵ')-non-malleable extractor with $\epsilon' = 2^{d-k'}\epsilon$.*[2]

Theorem 6.11. *[CRS11] For any constant $\delta > 0$, there is a constant c such that for any $\epsilon > 0$ there is an explicit $(k = (1/2 + \delta)n, \varepsilon)$-non-malleable extractor $\mathsf{nmExt} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ with $d = c(m + \log \varepsilon^{-1} + \log n)$.*

Now we have the following construction of non-malleable extractors.

[2] In fact, the non-malleable extractors in [DLWZ11, CRS11] can be shown to remain non-malleable even if the seed only contains a very small amount of min-entropy, but here we only discuss the general case.

Algorithm 6.12 (NMExt, non-malleable extractor with large output size)**.**

Input: x — an n bit string.
Output: z — an m bit string with $m = \Omega(n)$.

Sub-Routines and Parameters:
Let $\epsilon_1, \epsilon_2, \epsilon_3$ be three error parameters that we will choose later.
Let X be an (n, k)-source with $k = (1/2 + \delta)n$ for some constant $\delta > 0$.
Let DExt be an $(N, M = n, K, D, \alpha, \epsilon)$ design extractor as in Corollary 6.8, with $D = \Omega(n)$, error ϵ_1 and seed length $d_1 = \log N = O(\log \epsilon_1^{-1})$.
Let Ext be the optimal strong seeded extractor as in Theorem 3.9, with error ϵ_3 and output length $\delta D/10$.
Let nmExt be a non-malleable extractor as in Theorem 6.11 using $d_2 = 2d_1 + c(2d_1 + d_3 + \log \epsilon_2^{-1} + \log n)$ bits and outputting $d_3 = O(\log n + \log \epsilon_3^{-1})$ bits with error $\epsilon_2/2^{2d_1}$.

1. Associate the right hand side of DExt with the n bits of X. Choose a random seed Y_1 with d_1 bits and let $\bar{X} = \mathsf{DExt}(X, Y_1)$ be the output bits of X selected by Y_1.
2. Choose another independent random seed Y_2 with d_2 bits and compute $R = \mathsf{nmExt}(X, Y_2)$.
3. Output $Z = \mathsf{Ext}(\bar{X}, R) = \mathsf{Ext}(\mathsf{DExt}(X, Y_1), \mathsf{nmExt}(X, Y_2))$.

We have the following theorem.

Theorem 6.13. *The above construction is a (k, ϵ) non-malleable extractor with $k = (1/2 + \delta)n$, seed length $d = d_1 + d_2 + d_3 = O(\log n + \log \epsilon^{-1})$ and output length $m = \Omega(n)$.*

Proof. Without loss of generality assume that $\epsilon \leq 1/n$. Otherwise we can choose $\epsilon' = 1/n$ and $O(\log n + \log \epsilon'^{-1}) = O(\log n) = O(\log n + \log \epsilon^{-1})$. We will choose $\epsilon_1 < \epsilon, \epsilon_2 < \epsilon$ and $\epsilon_3 < \epsilon$. In the following we will use letters with prime to denote the corresponding random variables produced with $\mathcal{A}(Y)$. For example, $Y' = \mathcal{A}(Y)$.

Note that the seed of the extractor is $Y = (Y_1, Y_2)$. We want to show that

$$(\mathsf{NMExt}(X, Y), \mathsf{NMExt}(X, Y'), Y) \approx_\epsilon (U_m, \mathsf{NMExt}(X, Y'), Y).$$

for any deterministic function $Y' = \mathcal{A}(Y)$ and $Y' \neq Y$. Since $Y' \neq Y$, it must be that either $Y_1' \neq Y_1$ or $Y_2' \neq Y_2$. We consider the following three cases.

Case 1: $Y_1' = Y_1$ but $Y_2' \neq Y_2$. This case is relatively easy. We first fix $Y_1 = y_1$. By Lemma 6.4 and Lemma 6.5, we have

Claim 6.14. *With probability at least $1 - \epsilon_1$ over the fixing of Y_1, $\bar{X}$ is ϵ_1-close to a $(D, (1/2 + \delta - 3\tau)D > D/2)$ source.*

Now after this fixing $\bar{X} = \mathsf{DExt}(X, y_1)$ is a deterministic function of X and Y_2' is a deterministic function of Y_2. Note that $D < \alpha M < \delta n/10$. Thus we can now fix $\bar{X}$ and by Lemma 3.2 we have

Claim 6.15. *With probability $1 - 2^{-\Omega(\delta n)}$ over the fixings of $\bar{X}$, X has min-entropy at least $(1/2 + \delta/2)n$.*

Moreover after this fixing X and Y_2 are still independent. Now if conditioned on the fixing of $\bar{X}$, X indeed has min-entropy at least $(1/2+\delta/2)n$, then since Y_2' is now a deterministic function of Y_2, by Theorem 6.11 we have

$$(R, R', Y_2) \approx_{\epsilon_2} (U_{d_3}, R', Y_2).$$

Thus we have

Claim 6.16. *With probability $1-\epsilon_1$ over the fixings of Y_1, $\bar{X}$ is ϵ_1-close to a $(D, D/2)$ source, and*

$$|(\bar{X}, R, R', Y_2) - (\bar{X}, U_{d_3}, R', Y_2)| \leq 2^{-\Omega(\delta n)} + \epsilon_2,$$

where U_{d_3} is the uniform distribution over $\{0,1\}^{d_3}$ and independent of $(\bar{X}, R', Y_2)$.

Now we consider the distribution $(\bar{X}, R, R', Y_2)$ where R is actually equal to U_{d_3} and independent of $(\bar{X}, R', Y_2)$. This will increase the error by at most $2^{-\Omega(\delta n)}+\epsilon_2$. Now we fix Y_2, R' and $\mathsf{Ext}(\bar{X}, R')$. Note that after this fixing R is still uniform and independent of $\bar{X}$. Furthermore since the size of $(\mathsf{Ext}(\bar{X}, R'), Y_2, R')$ is at most $\delta D/10 + d_2 + d_3 < \delta D/4$. Thus by Lemma 3.2 we have

Claim 6.17. *With probability $1-2^{-\Omega(D)}$ over the fixings of $(\mathsf{Ext}(\bar{X}, R'), Y_2, R')$, $\bar{X}$ is a $(D, D/2 - \delta D/4 - D/8 > D/4)$ source.*

When $\bar{X}$ has min-entropy at least $D/4$, since R is independent of $\bar{X}$, by the property of the strong extractor Ext, we have

$$|\mathsf{Ext}(\bar{X}, R) - U_m| \leq \epsilon_3.$$

Thus by combining Claim 6.14, Claim 6.16 and Claim 6.17 we have

$$|(Z, Z', Y) - (U_m, Z', Y)| \leq \epsilon_1 + \epsilon_1 + 2^{-\Omega(\delta n)} + \epsilon_2 + 2^{-\Omega(D)} + \epsilon_3 = O(\epsilon_1 + \epsilon_2 + \epsilon_3 + 2^{-\Omega(n)}).$$

Case 2. $Y_1' \neq Y_1$ but $Y_2' = Y_2$. Again, we first fix $Y_1 = y_1$. By Corollary 6.8, with probability at least $1-\epsilon_1$ over this fixing, no matter what Y_1' is, with probability at least $1-\epsilon_1$ over the fixing of $\bar{X}'$, $\bar{X}$ is ϵ_1-close to a $\delta D/2$ source. Thus we can now fix $Y_1' = y_1'$ (note that Y_1' may not be a deterministic function of Y_1 since it can also depend on Y_2), and we have

Claim 6.18. *With probability at least $1-\epsilon_1$ over the fixing of $\bar{X}'$, $\bar{X}$ is ϵ_1-close to a $\delta D/2$ source.*

Note that we first fix $Y_1 = y_1$. After this fixing Y_1' is a deterministic function of Y_2. Since Y_1' has size d_1 and Y_2 has size $d_2 = 2d_1 + c(2d_1 + d_3 + \log \epsilon_2^{-1} + \log n)$, by Lemma 3.2 we have

Claim 6.19. *With probability at least $1-\epsilon_1$ over the fixing of Y_1', Y_2 is a $(d_2, c(2d_1 + d_3 + \log \epsilon_2^{-1} + \log n))$ source.*

Now after we fixed Y_1 and Y_1', $\bar{X}$ and $\bar{X}'$ are deterministic functions of X. Thus we can fix them and since the size of $(\bar{X}, \bar{X}')$ is at most $2D < \delta n/5$, by Lemma 3.2 we have

Claim 6.20. *With probability at least $1-2^{-\Omega(\delta n)}$ over the fixings of $(\bar{X}, \bar{X}')$, X is a $(n, (1/2+\delta/2)n)$ source.*

Note that after all these fixings Y_2 and X are still independent. Since a non-malleable extractor is also a strong extractor (even with weak seed), by Lemma 6.10 and Theorem 6.11 we have

$$|(R, \bar{X}, \bar{X}', Y_1', Y_2) - (U_{d_3}, \bar{X}, \bar{X}', Y_1', Y_2)| \leq \epsilon_1 + 2^{-\Omega(\delta n)} + 2^{2d_1}\epsilon_1/2^{2d_1} = \epsilon_1 + \epsilon_2 + 2^{-\Omega(\delta n)}.$$

Note that here we are conditioning on the fixing of $Y_1 = y_1$, thus Y_1', Y_2 are independent of $\bar{X}$. Thus by the property of the strong extractor, together with Claim 6.18 we have

$$|(Z, Z', Y_1', Y_2, R) - (U_m, Z', Y_1', Y_2, R)| \leq \epsilon_1 + \epsilon_1 + \epsilon_2 + 2^{-\Omega(\delta n)} + \epsilon_3 = 2\epsilon_1 + \epsilon_2 + \epsilon_3 + 2^{-\Omega(\delta n)}.$$

Adding back the probability of a bad $Y_1 = y_1$, we have

$$|(Z, Z', Y) - (U, Z', Y)| \leq 3\epsilon_1 + \epsilon_2 + \epsilon_3 + 2^{-\Omega(\delta n)} = O(\epsilon_1 + \epsilon_2 + \epsilon_3 + 2^{-\Omega(n)}).$$

Case 3 $Y_1' \neq Y_1$ and $Y_2' \neq Y_2$. The first part of the case is the same as **Case 2**. Specifically, we first fix $Y_1 = y_1$. By Corollary 6.8, with probability at least $1 - \epsilon_1$ over the fixings, no matter what Y_1' is, with probability at least $1 - \epsilon_1$ over the fixings of $\bar{X}'$, $\bar{X}$ is ϵ_1-close to a $\delta D/2$ source. Thus we can now fix $Y_1' = y_1'$ (note that Y_1' may not be a deterministic function of Y_1 since it can also depend on Y_2), and we have

Claim 6.21. *With probability at least $1 - \epsilon_1$ over the fixing of $\bar{X}'$, $\bar{X}$ is ϵ_1-close to a $\delta D/2$ source.*

Note that we first fix $Y_1 = y_1$. After this fixing Y_1' is a deterministic function of Y_2. Since Y_1' has size d_1 and Y_2 has size $d_2 = 2d_1 + c(2d_1 + d_3 + \log \epsilon_2^{-1} + \log n)$, by Lemma 3.2 we have

Claim 6.22. *With probability at least $1 - \epsilon_1$ over the fixing of Y_1', Y_2 is a $(d_2, c(2d_1 + d_3 + \log \epsilon_2^{-1} + \log n))$ source.*

Now after we fixed Y_1 and Y_1', $\bar{X}$ and $\bar{X}'$ are deterministic functions of X. Thus we can fix them and since the size of $(\bar{X}, \bar{X}')$ is at most $2D < \delta n/5$, by Lemma 3.2 we have

Claim 6.23. *With probability at least $1 - 2^{-\Omega(\delta n)}$ over the fixings of $(\bar{X}, \bar{X}')$, X is a $(n, (1/2 + \delta/2)n)$ source.*

From here the proof differs from **Case 2**. Note that after all these fixings Y_2 and X are still independent, and now Y_2' is a deterministic function of Y_2. Thus by Lemma 6.10 and Theorem 6.11 we have

$$|(R, R', \bar{X}, \bar{X}', Y_1', Y_2, Y_2') - (U_{d_3}, R', \bar{X}, \bar{X}', Y_1', Y_2, Y_2')| \leq 2\epsilon_1 + 2^{-\Omega(\delta n)} + 2^{2d_1}\epsilon_2/2^{2d_1} = 2\epsilon_1 + \epsilon_2 + 2^{-\Omega(\delta n)}.$$

Note that here we are conditioning on the fixing of $Y_1 = y_1$, thus Y_1', Y_2, Y_2' are independent of $\bar{X}$ and we can fix them. R' may be correlated with $\bar{X}$. However, note that R' has size at most $d_3 = O(\log n + \log \epsilon_3^{-1})$. Thus by Lemma 3.2 we have

Claim 6.24. *With probability at least $1 - 2^{-\Omega(\delta D)}$ over the fixings of R', $\bar{X}$ is a $(n, \delta D/4)$ source.*

Now we can fix $(R', \bar{X}')$. After this fixing R is still close to uniform and independent of $\bar{X}$. Moreover $\bar{X}$ still has a lot of min-entropy. On the other hand after this fixing $Z' = \mathsf{Ext}(\bar{X}', R')$ is fixed. Therefore by the property of the strong extractor we have

$$|(Z, Z', Y, Y') - (U_m, Z', Y, Y')| \leq \epsilon_3 + 2^{-\Omega(\delta D)} + 2\epsilon_1 + \epsilon_2 + 2^{-\Omega(\delta n)} = 2\epsilon_1 + \epsilon_2 + \epsilon_3 + 2^{-\Omega(n)}.$$

Adding back the probability of a bad $Y_1 = y_1$, we have

$$|(Z, Z', Y) - (U_m, Z', Y)| \leq 3\epsilon_1 + \epsilon_2 + \epsilon_3 + 2^{-\Omega(n)} = O(\epsilon_1 + \epsilon_2 + \epsilon_3 + 2^{-\Omega(n)}).$$

Note that the total error achieved by an adversary $\mathcal{A}$ is at most the sum of the errors in the above three cases, thus

$$|(Z, Z', Y) - (U_m, Z', Y)| \leq O(\epsilon_1 + \epsilon_2 + \epsilon_3 + 2^{-\Omega(n)}).$$

By choosing $\epsilon_1, \epsilon_2, \epsilon_3 = \Theta(\epsilon)$ appropriately, we have that

$$|(Z, Z', Y) - (U_m, Z', Y)| \leq \epsilon,$$

the seed length is $d = O(\log n + \log \epsilon^{-1})$, and the output length is $m = \Omega(n)$. ■

7 Non-Malleable Condensers

In this section, we show that the construction we give in the previous section already gives a *non-malleable condenser*. Recall the definition of a non-malleable condenser.

Definition 7.1. A (k, k', ϵ) non-malleable condenser is a function $\mathsf{nmCond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ such that given any (n,k)-source X, an independent uniform seed $Y \in \{0,1\}^d$, and any (deterministic) function $\mathcal{A} : \{0,1\}^d \to \{0,1\}^d$ such that $\forall y, \mathcal{A}(y) \neq y$, we have that with probability $1 - \epsilon$ over the fixings of $Y = y$,

$$\Pr_{z' \leftarrow \mathsf{nmCond}(X, \mathcal{A}(y))}[\mathsf{nmCond}(X, y)|_{\mathsf{nmCond}(X, \mathcal{A}(y)) = z'} \text{ is } \epsilon - \text{close to an } (m, k') \text{ source}] \geq 1 - \epsilon.$$

Similar as the fact that an extractor is a special case and a stronger version of a condenser, a non-malleable extractor is also a special case and a stronger version of a non-malleable condenser. Indeed, a (k, ϵ) non-malleable extractor with output size m is just a $(k, m, \sqrt[3]{\epsilon})$ non-malleable condenser. Thus a non-malleable condenser is a strictly weaker object than a non-malleable extractor. Dodis and Wichs showed that non-malleable extractors exist with $k > 2m + 3\log(1/\varepsilon) + \log d + 9$ and $d > \log(n - k + 1) + 2\log(1/\varepsilon) + 7$, thus non-malleable condensers exist with at least these parameters.

It can be seen easily from Corollary 6.8 that when X has min-entropy $k \geq (1/2 + \delta)n$ for any constant $\delta > 0$, the design extractor actually gives a non-malleable condenser for X. Indeed, Corollary 6.8 gives the following theorem.

Theorem 7.2. *For any constant $\delta > 0$, there exists a constant $\beta > 0$ such that for any $n, k \in \mathbb{N}$ with $k \geq (1/2 + \delta)n$ and $\epsilon > 2^{-\beta n}$, there exists an efficiently computable (k, k', ϵ) non-malleable condenser with $d = O(\log(1/\epsilon)), m = \Omega(n)$ and $k' \geq \delta m/2$.*

We now give two applications of non-malleable condensers below.

7.1 From non-malleable condenser to non-malleable extractor

In the context of randomness extractors, condensers are used as an intermediate object to construct extractors. Namely, we first use condensers to convert a weak source with low min-entropy rate into another source with high min-entropy rate, where randomness extraction becomes easier. Here we show that non-malleable condensers can also be used to construct non-malleable extractors. For example, we have the following theorem.

Theorem 7.3. *Assume we have an explicit (k, k', ϵ) non-malleable condenser $\mathsf{nmCond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ such that $k' \geq (1/2+\delta)m$ for any constant $\delta > 0$ and $2^d \geq 1/\epsilon$. Then there exists an efficiently computable $(k, 9\epsilon)$ non-malleable extractor with seed length $O(d + \log m + \log \epsilon^{-1})$ and output length $\Omega(m)$.*

Before proving this theorem, we need an alternative description of the non-malleable condenser.

Lemma 7.4. *Assume that we have a (k, k', ϵ) non-malleable condenser $\mathsf{nmCond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$. Given any (n,k)-source X, for at least $1-\epsilon$ fraction of choices $y \in \{0,1\}^d$, we have that $\forall y' \neq y$,*

$$\Pr_{z' \leftarrow \mathsf{nmCond}(X,y')}[\mathsf{nmCond}(X,y)|_{\mathsf{nmCond}(X,y')=z'} \text{ is } \epsilon - \text{close to an } (m,k') \text{ source}] \geq 1-\epsilon.$$

Note the difference between the above statement and the definition of a non-malleable condenser. The definition says that given any $\mathcal{A}$ such that $\forall y, \mathcal{A}(y) \neq y$, there is a $1-\epsilon$ fraction of good seeds y. The lemma above says that there is a $1-\epsilon$ fraction of good seeds y regardless of what the function $\mathcal{A}$ is. We show that the definition implies the lemma (In fact, these statements are equivalent).

Proof. Assume for the sake of contradiction that the lemma does not hold. Thus, we have that for some (n,k)-source X, there is an $S \subset \{0,1\}^d$, $|S| \geq \epsilon 2^d$, such that $\forall y \in S$, there exists $y' \in \{0,1\}^d$, $y' \neq y$ and

$$\Pr_{z' \leftarrow \mathsf{nmCond}(X,y')}[\mathsf{nmCond}(X,y)|_{\mathsf{nmCond}(X,y')=z'} \text{ is } \epsilon - \text{close to an } (m,k') \text{ source}] < 1-\epsilon.$$

Now, we can take a function $\mathcal{A}$ such that $\forall y \in S, \mathcal{A}(y) = y'$ and let $\mathcal{A}(y)$ be arbitrary for all the other y's. The source X and the function $\mathcal{A}$ now contradict the definition of the non-malleable condenser. Thus the lemma is proved. □

We can now prove the theorem.

Proof of Theorem 7.3. The construction of the non-malleable extractor is simple. Just take the non-malleable extractor nmExt from Theorem 6.13 and combine it with the non-malleable condenser. Specifically, we take two independent uniform seed Y_1, Y_2, where Y_1 has d bits and Y_2 has $d_2 = O(d + \log m + \log \epsilon^{-1})$ bits, and the output is

$$Z = \mathsf{nmExt}(\mathsf{nmCond}(X, Y_1), Y_2).$$

We now show that the construction is a non-malleable extractor for (n,k)-sources. Let $W = \mathsf{nmCond}(X, Y_1)$. As usual, we will use letters with prime to denote the corresponding random

variables produced by using $\mathcal{A}(Y)$ instead of Y. Similarly, since $Y = Y_1 \circ Y_2$ and $\mathcal{A}(Y) \neq Y$, we have the following two cases.

Case 1: $Y_1' = Y_1$ but $Y_2' \neq Y_2$. We first condition on $Y_1 = y_1$. By Lemma 7.4, with probability $1 - \epsilon$ over the fixings of $Y_1 = y_1$, $W = \mathsf{nmCond}(X, y_1)$ is 2ϵ-close to an (m, k')-source.

Now after we fix $Y_1 = y_1$, W is a deterministic function of X and Y_2' is a deterministic function of Y_2. Thus W and Y_2 are independent. Note that $k' \geq (1/2 + \delta)m$, therefore by Theorem 6.13 we have

$$|(Z, Z', Y_2) - (U, Z', Y_2)| \leq 2\epsilon + \epsilon = 3\epsilon.$$

Adding back the error ϵ of conditioning on $Y_1 = y_1$, we have

$$|(Z, Z', Y) - (U, Z', Y)| \leq 3\epsilon + \epsilon = 4\epsilon.$$

Case 2: $Y_1' \neq Y_1$. Again, we first condition on $Y_1 = y_1$. By Lemma 7.4, with probability $1 - \epsilon$ over the fixing of $Y_1 = y_1$, no matter what y_1' is, W conditioned on $W' = w'$ has high min-entropy. Thus we further condition on $Y_1' = y_1'$ and $W' = w'$, and we have

$$\Pr_{w' \leftarrow W'}[W|_{W'=w'} \text{ is } \epsilon - \text{close to an } (m, k') \text{ source}] \geq 1 - \epsilon.$$

Note that we first conditioned on $Y_1 = y_1$. After this fixing Y_1' is a deterministic function of Y_2. Next we fix $Y_1' = y_1'$. Since Y_1' has d bits and Y_2 has d_2 bits, by Lemma 3.2 we have that with probability $1 - 2^{-d} \geq 1 - \epsilon$ over the fixing of $Y_1' = y_1'$, Y_2 has min-entropy at least $d_2 - d - d = d_2 - 2d$. We choose $d_2 = O(d + \log m + \log \epsilon^{-1})$ such that the error of nmExt in Theorem 6.13 is at most $\epsilon/2^{2d}$. Thus when Y_2 has min-entropy at least $d_2 - 2d$, by Lemma 6.10 (and noticing that a non-malleable extractor is also a strong extractor) we have

$$|(Z, Y_2) - (U, Y_2)| \leq \epsilon + \epsilon = 2\epsilon.$$

Note that the above inequality is conditioning on the fixing of $Y_1 = y_1, Y_1' = y_1', W' = w'$. After this fixing Y_2' is a deterministic function of Y_2. Thus we also have

$$|(Z, Y_2, Y_2') - (U, Y_2, Y_2')| \leq \epsilon + \epsilon = 2\epsilon.$$

Adding back all the errors we get

$$|(Z, W', Y_1, Y_1', Y_2, Y_2') - (U, W', Y_1, Y_1', Y_2, Y_2')| \leq 2\epsilon + \epsilon + \epsilon + \epsilon = 5\epsilon.$$

Since $Z' = \mathsf{nmExt}(W', Y_2')$, we have

$$|(Z, Z', Y) - (U, Z', Y)| \leq 5\epsilon.$$

Note that the total error achieved by an adversary $\mathcal{A}$ is at most the sum of the errors in the two cases, therefore the construction is a $(k, 9\epsilon)$-non-malleable extractor with output length $\Omega(m)$. ■

7.2 A new two round optimal protocol for privacy amplification

In [DW09], Dodis and Wichs showed that non-malleable extractors give optimal two round protocols for privacy amplification with an active adversary. One can ask the natural question of whether the reverse is true, i.e., does an optimal two round protocol for privacy amplification essentially require a non-malleable extractor? Here we show that this is not necessary. In fact, we show that it suffices to use a non-malleable condenser, which is much weaker than a non-malleable extractor. First we have the following claim.

Claim 7.5. *For every constant $c > 1$ there exists a constant $0 < \beta < 1$ such that for any n, k and $2^{-\beta k} < \epsilon < 1$ there is a (possibly non-explicit) (k, k', ϵ)-non-malleable condenser* $\mathsf{nmCond} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *with $k > 2m + 3\log(1/\varepsilon) + \log d + 9$, $k' \geq c(\log n + \log(1/\epsilon))$ and $d = O(\log n + \log(1/\epsilon))$.*

Proof. This follows directly from the existence proof of non-malleable extractors in [DW09] and the fact that a non-malleable extractor is also a non-malleable condenser. □

We now formally define a privacy amplification protocol with an active adversary. Following [KR09], let $x \in \{0,1\}^n$ be the secret string shared by Alice and Bob, where x is sampled according to a distribution X. Let Protocol (P_A, P_B) be executed in the presence of an active adversary Eve. Let V_a denote the random variable that describes Alice's view of the communication when (P_A, P_B) is executed and define V_b likewise. We use small letters v_a, v_b to denote specific values of V_a, V_b. The private randomness of Alice and Bob are denoted by y and w respectively. Alice's output is denoted by $r_A = P_A(x, v_a, y)$ and Bob's output is denoted by $r_B = P_B(x, v_b, w)$ (if successful, both outputs will be of length m; rejection will be denoted by symbol $\perp$). Let V denote Eve's view of the protocol. Since Eve is computationally unbounded, we can simply assume that Eve is deterministic.

Definition 7.6. [KR09] An interactive protocol (P_A, P_B), executed by Alice and Bob on a communication channel fully controlled by an active adversary Eve, is a (k, m, ϵ)-*privacy amplification protocol* if it satisfies the following properties whenever $H_\infty(X) \geq k$:

1. Correctness. If Eve is passive, then $\Pr[R_A = R_B] = 1$.
2. Robustness. The probability that the following experiment outputs "Eve wins" is at most ϵ: sample x from X; let v_a, v_b be the communication upon execution of (P_A, P_B) with Eve actively controlling the channel, and let $r_A = P_A(x, v_a, y)$, $r_B = P_B(x, v_b, w)$. Output "Eve wins" if $(r_A \neq r_B \wedge r_A \neq \perp \wedge r_B \neq \perp)$.
3. Extraction. Letting V denote Eve's view of the protocol,

$$|(R_A, V | R_A \neq \perp) - (U_m, V)| \leq \epsilon$$

and

$$|(R_B, V | R_B \neq \perp) - (U_m, V)| \leq \epsilon.$$

Here $s = \log(1/\epsilon)$ is called the *security parameter* of the protocol, and $k - m$ is called the *entropy loss* of the protocol.

7.2.1 Prerequisites from previous work

One-time message authentication codes (MACs) use a shared random key to authenticate a message in the information-theoretic setting.

Definition 7.7. A function family $\{\mathsf{MAC}_R : \{0,1\}^d \to \{0,1\}^v\}$ is a ϵ-secure one-time MAC for messages of length d with tags of length v if for any $w \in \{0,1\}^d$ and any function (adversary) $A : \{0,1\}^v \to \{0,1\}^d \times \{0,1\}^v$,

$$\Pr_R[\mathsf{MAC}_R(W') = T' \wedge W' \neq w \mid (W', T') = A(\mathsf{MAC}_R(w))] \leq \epsilon,$$

where R is the uniform distribution over the key space $\{0,1\}^\ell$.

Theorem 7.8 ([KR09])**.** *For any message length d and tag length v, there exists an efficient family of $(\lceil \frac{d}{v} \rceil 2^{-v})$-secure* MAC*s with key length $\ell = 2v$. In particular, this* MAC *is ε-secure when $v = \log d + \log(1/\epsilon)$.*
More generally, this MAC *also enjoys the following security guarantee, even if Eve has partial information E about its key R. Let (R, E) be any joint distribution. Then, for all attackers A_1 and A_2,*

$$\Pr_{(R,E)}[\mathsf{MAC}_R(W') = T' \wedge W' \neq W \mid W = A_1(E),\ (W', T') = A_2(\mathsf{MAC}_R(W), E)] \leq \left\lceil \frac{d}{v} \right\rceil 2^{v - \widetilde{H}_\infty(R|E)}.$$

(In the special case when $R \equiv U_{2v}$ and independent of E, we get the original bound.)

Remark 7.9. Note that the above theorem indicates that the MAC works even if the key R has average min-entropy rate $> 1/2$.

7.2.2 The protocol

Now we give our privacy amplification protocol. We assume that the shared weak random source has min-entropy k, and the error ϵ we seek satisfies $2^{-\Omega(\beta k)} < \epsilon < 1/n$. For convenience, in the description below we introduce an "auxiliary" security parameter s. Eventually, we will set $s = \log(C/\epsilon) + O(1) = \log(1/\epsilon) + O(1)$, so that $C/2^s < \epsilon$, for a sufficiently large constant C related to the number of "bad" events we need to account for. We need the following building blocks:

- Let $\mathsf{nmCond} : \{0,1\}^n \times \{0,1\}^{d_1} \to \{0,1\}^l$ be the non-malleable condenser from Claim 7.5, with $l = \Omega(k)$ and error 2^{-s}.
- Let $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^{d_1} \to \{0,1\}^m$ be a $(k, 2^{-s})$-extractor with optimal entropy loss $k - m = O(s)$.
- Let $\mathsf{Raz} : \{0,1\}^{d_2} \times \{0,1\}^l \to \{0,1\}^t$ be the two source extractor from Theorem 3.10, with $d_2 \geq 5d_1$, error 2^{-s} and $t = s$.
- Let MAC be the ("leakage-resilient") MAC for d_1-bit messages, as in Theorem 7.8, with tag length $v = 2s$ and key length $\ell = 2v = 4s$.

Using the above building blocks, the protocol is given in Figure 1. To emphasize the presence of Eve, we use letters with 'prime' to denote all the protocol values seen or generated by Bob; e.g., Bob picks W_1', but Alice sees potentially different W_1, etc. Also, for any random variable G used in describing our protocol, we use the notation $G = \perp$ to indicate that G was never assigned any value, because the party who was supposed to assign G rejected earlier. The case of final keys R_A and R_B becomes a special case of this convention.

Alice: X	Eve: E	Bob: X
Sample random Y_1, Y_2. Compute $Z = \mathsf{nmCond}(X, Y_1)$. $R = \mathsf{Ext}(X, Y_1)$ and output $4s$ bits.	$(Y_1, Y_2) \longrightarrow (Y_1', Y_2')$	
		Sample random W'. Compute $Z' = \mathsf{nmCond}(X, Y_1')$. $R' = \mathsf{Ext}(X, Y_1')$ and output $4s$ bits. $T_1' = \mathsf{Raz}(Y_2', Z')$, $T_2' = \mathsf{MAC}_{R'}(W')$. Set final $R_B = \mathsf{Ext}(X, W')$.
	$(W, T_1, T_2) \longleftarrow (W', T_1', T_2')$	
If $T_1 \neq \mathsf{Raz}(Y_2, Z)$ or $T_2 \neq \mathsf{MAC}_R(W)$ *reject.* Set final $R_A = \mathsf{Ext}(X, W)$.		

Figure 1: 2-round Privacy Amplification Protocol.

Theorem 7.10. *The above protocol is a privacy amplification protocol with security parameter* $\log(1/\epsilon)$ *and entropy loss* $O(\log(1/\epsilon))$*, communication complexity* $O(\log(1/\epsilon))$*.*

Proof. The proof can be divided into two cases: whether the adversary changes Y_1 or not. Note that Y_1, Y_2 and W all have size $O(s)$.

Case 1: The adversary does not change Y_1. In this case, note that $R = R'$ and is 2^{-s}-close to uniform in Eve's view (even conditioned on Y_1, Y_2). Thus the property of the MAC guarantees that Bob can authenticate W' to Alice. However, one thing to note here is that Eve has some additional information, namely T_1' which can leak information about the MAC key. On the other hand, the size of T_1' is s, thus by Lemma 3.7 the average conditional min-entropy $H_\infty(R|T_1')$ is at least $3s$. Therefore by Theorem 7.8 the probability that Eve can change W' to a different W without causing Alice to reject is at most

$$\left\lceil \frac{d_1}{2s} \right\rceil 2^{2s - \widetilde{H}_\infty(R|T_1')} + 2^{-s} \leq O(2^{2s-3s}) + 2^{-s} \leq O(2^{-s}).$$

When $W = W'$, by Theorem 3.9 $R_A = R_B$ and is 2^{-s}-close to uniform in Eve's view.

Case 2: The adversary does change Y_1. In this case, by the property of the non-malleable condenser, with probability $1 - 2^{-s}$ over the fixing of $Y_1 = y_1$,

$$\Pr_{z' \leftarrow \mathsf{nmCond}(X, y_1')} [\mathsf{nmCond}(X, y_1)|_{\mathsf{nmCond}(X, y_1') = z'} \text{ is } 2^{-s} - \text{close to an } (l, k') \text{ source}] \geq 1 - 2^{-s}.$$

Thus we first fix $Y_1 = y_1$ and then fix $Y_1' = y_1'$. Note that after we fix Y_1, Y_1' is a deterministic function of Y_2. Since Y_1 has d_1 bits and Y_2 has $d_2 \geq 5d_1$ bits, we can fix $Y_1' = y_1'$ and by Lemma 3.2, with probability $1 - 2^{-d_1} \geq 1 - 2^{-s}$ over this fixing, Y_2 has min-entropy $d_2 - d_1 - d_1 \geq 3d_1$.

Next we fix $Z' = \mathsf{nmCond}(X, y_1')$ and we know that with probability $1 - 2^{-s}$ over this fixing, $Z = \mathsf{nmCond}(X, y_1)$ is 2^{-s}-close to having min-entropy $k' \geq c(\log n + s) > cs$. Note that W' is independent of everything else, thus we can fix W' and now T_2' is a deterministic function of X. Since T_2' has size at most $2s$, we can now fix T_2' and by Lemma 3.2 with probability $1 - 2^{-s}$ over the fixings, Z is 2^{-s}-close to having min-entropy $k' - 3s > (c-3)s$.

Note that now we have fixed Y_1, Y_1', Z', T_2'. Further note now that Y_2 and Z are still independent, since Z is a deterministic function of X, and X is independent of Y_2. Note that Y_2 has min-entropy at least $d_2 - 2d_1 > d_2/2$. Let $\bar{T}_1 = \mathsf{Raz}(Y_2, Z)$, by Theorem 3.10, with an appropriately chosen $d_2 = O(s)$ and sufficiently large constant c we have that we can output s bits in $\bar{T}_1$ and

$$(\bar{T}_1, Y_2) \approx_{2^{-s}} (U, Y_2).$$

Note that now after all the fixings, Y_2' is a deterministic function of Y_2, and the error introduced by the fixings is at most $O(2^{-s})$. Thus

$$(\bar{T}_1, Y_1, Y_1', Y_2, Y_2', Z', W', T_2') \approx_{O(2^{-s})} (U, Y_1, Y_1', Y_2, Y_2', Z', W', T_2').$$

Note that $T_1' = \mathsf{Raz}(Y_2', Z')$, thus

$$(\bar{T}_1, Y, T_1', W', T_2') \approx_{O(2^{-s})} (U, Y, T_1', W', T_2').$$

Therefore, the probability that Eve can come up with the correct $T_1 = \bar{T}_1$ for Alice is at most $2^{-s} + O(2^{-s}) = O(2^{-s})$. The total error probability that Eve can achieve is at most the sum of the above two cases, which is $O(2^{-s})$. For an appropriately chosen $s = O(\log(1/\epsilon))$ this is at most ϵ.

Finally note that every string transmitted has size $O(s)$, thus the entropy loss is $O(s) = O(\log(1/\epsilon))$, and the communication complexity is also $O(s) = O(\log(1/\epsilon))$. ■

Plugging in Theorem 7.2, we obtain a 2-round protocol with optimal entropy loss for min-entropy $k \geq (1/2 + \delta)n$, without using non-malleable extractors.

Theorem 7.11. *There exists a constant $0 < \beta < 1$ such that for any constant $\delta > 0$, $k = (1/2+\delta)n$ and $\epsilon > 2^{-\beta\delta n}$, there exists an explicit privacy amplification protocol for (n, k)-sources with security parameter $\log(1/\epsilon)$, entropy loss $O(\log(1/\epsilon))$ and communication complexity $O(\log(1/\epsilon))$, in the presence of an active adversary.*

7.3 Non-malleable condensers and MACs

Here we point out a relation between non-malleable condensers and MACs (message authentication codes). We have the following claim.

Claim 7.12. *A ϵ-secure one time MAC that works when the key R is any (n, k)-source is a $(k, k', \sqrt{\epsilon})$ non-malleable condenser with $k' = \frac{1}{2}\log(1/\epsilon)$.*

Proof. In the definition of the MAC (Definition 7.7), we view the message W as the seed of the non-malleable condenser, and the tag $T = MAC_R(W)$ as the output of the non-malleable condenser. For any fixed message w, we let the function A in Definition 7.7 be defined as follows.

For any fixed $t = MAC_R(w)$, consider the set of distributions $\{MAC_R(w')|_{MAC_R(w)=t}, w' \neq w\}$. Let $p_t = max_{w',t'} \Pr[MAC_R(w')|_{MAC_R(w)=t} = t']$ and (w_1, t_1) be the pair where p is achieved. Now let $A(t) = (w_1, t_1)$. Thus we have that

$$\Pr_R[\mathsf{MAC}_R(W') = T' \wedge W' \neq w \mid (W', T') = A(\mathsf{MAC}_R(w))] = \sum_t \Pr[T = t]p_t = E_T[p_t].$$

By definition we have that $E_T[p_t] \leq \epsilon$. Thus by Markov's inequality we have

$$\Pr_T[p_t \leq \sqrt{\epsilon}] \geq 1 - \sqrt{\epsilon}.$$

Therefore, for any seed w, with probability $1 - \sqrt{\epsilon}$ over the fixing of $T = MAC_R(w)$, for any other seed $w' \neq w$, $MAC_R(w')$ has min-entropy at least $\frac{1}{2}\log(1/\epsilon)$ (since the probability mass of any element in the support is at most $\sqrt{\epsilon}$). Thus this is a a $(k, k', \sqrt{\epsilon})$ non-malleable condenser with $k' = \frac{1}{2}\log(1/\epsilon)$. □

Note that the above proof shows that the MAC in fact gives something stronger, in the sense that for any two different w, w', with probability $1 - \sqrt{\epsilon}$ over the fixing of $T = MAC_R(w)$, $MAC_R(w')$ has min-entropy at least $\frac{1}{2}\log(1/\epsilon)$. Thus the reverse is not true. In particular, a non-malleable condenser may not be a MAC, because there might be a bad seed w such that $\mathsf{nmCond}(X, w)$ is fixed and thus it would be easy for an adversary to change a w' to w.

Given this relation, Theorem 7.8 thus also gives a non-malleable condenser for (n, k)-sources with $k > n/2$. However, this construction has a large seed length, i.e., the seed length $d \geq n/2$. Also, the output size must be at least $n/2$. Thus it is not suitable for our applications. On the other hand, our non-malleable condenser has seed length $d = O(\log(1/\epsilon))$, thus we achieve an optimal seed length with respect to the error, and we can adjust our seed length and output size appropriately according to the applications.

8 Conclusions and Open Problems

In this paper we introduce a new combinatorial object called a design extractor, that is both a design and an extractor. We give efficient constructions of design extractors and use them to improve the output length of known non-malleable extractors. We then introduce the notion of a non-malleable condenser and show that our design extractor gives a non-malleable condenser for min-entropy $k > n/2$. We show that non-malleable condensers can be used to construct non-malleable extractors, and non-malleable condensers alone can be used to give optimal privacy amplification protocols with an active adversary.

There are several natural open problems left. The first is to construct non-malleable condensers for smaller min-entropy, so that we can obtain better non-malleable extractors and better privacy amplification protocols. This approach seems promising. Another problem is to find new applications of our design extractors.

Acknowledgments

We would like to thank David Zuckerman for many useful discussions, especially for his suggestion to use Trevisan's extractor.

References

[CKOR10] N. Chandran, B. Kanukurthi, R. Ostrovsky, and L. Reyzin. Privacy amplification with asymptotically optimal entropy loss. In *Proceedings of the 42nd Annual ACM Symposium on Theory of Computing*, pages 785–794, 2010.

[CRS11] Gil Cohen, Ran Raz, and Gil Segev. Non-malleable extractors with short seeds and applications to privacy amplification. Technical Report TR11-096, ECCC, 2011.

[DKRS06] Y. Dodis, J. Katz, L. Reyzin, and A. Smith. Robust fuzzy extractors and authenticated key agreement from close secrets. In *CRYPTO*, pages 232–250, 2006.

[DKSS09] Zeev Dvir, Swastik Kopparty, Shubhangi Saraf, and Madhu Sudan. Extensions to the method of multiplicities, with applications to kakeya sets and mergers. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science*, 2009.

[DLWZ11] Yevgeniy Dodis, Xin Li, Trevor D. Wooley, and David Zuckerman. Privacy amplification and non-malleable extractors via character sums. In *Proceedings of the 52nd Annual IEEE Symposium on Foundations of Computer Science*, 2011.

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

[DW08] Zeev Dvir and Avi Wigderson. Kakeya sets, new mergers and old extractors. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, 2008.

[DW09] Yevgeniy Dodis and Daniel Wichs. Non-malleable extractors and symmetric key cryptography from weak secrets. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, page 601610, 2009.

[GS00] V. Guruswami and M. Sudan. List decoding algorithms for certain concatenated codes. In *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, pages 181–190, 2000.

[GUV09] Venkatesan Guruswami, Christopher Umans, and Salil Vadhan. Unbalanced expanders and randomness extractors from Parvaresh-Vardy codes. *Journal of the ACM*, 56(4), 2009.

[KR09] B. Kanukurthi and L. Reyzin. Key agreement from close secrets over unsecured channels. In *EUROCRYPT*, pages 206–223, 2009.

[LRVW03] C. J. Lu, Omer Reingold, Salil Vadhan, and Avi Wigderson. Extractors: Optimal up to constant factors. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pages 602–611, 2003.

[MW97] Ueli M. Maurer and Stefan Wolf. Privacy amplification secure against active adversaries. In *CRYPTO '97*, 1997.

[NW94] Noam Nisan and Avi Wigderson. Hardness vs randomness. *Journal of Computer and System Sciences*, 49(2):149–167, October 1994.

[NZ96] Noam Nisan and David Zuckerman. Randomness is linear in space. *Journal of Computer and System Sciences*, 52(1):43–52, 1996.

[Raz05] Ran Raz. Extractors with weak random seeds. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 11–20, 2005.

[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.

[RW03] R. Renner and S. Wolf. Unconditional authenticity and privacy from an arbitrarily weak secret. In *CRYPTO*, pages 78–95, 2003.

[SZ99] Aravind Srinivasan and David Zuckerman. Computing with very weak random sources. *SIAM Journal on Computing*, 28:1433–1459, 1999.

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

[Vad04] Salil P. Vadhan. On constructing locally computable extractors and cryptosystems in the bounded-storage model. *J. Cryptology*, 17(1):43–77, 2004.

[WZ99] Avi Wigderson and David Zuckerman. Expanders that beat the eigenvalue bound: Explicit construction and applications. *Combinatorica*, 19(1):125–138, 1999.

[Zuc97] David Zuckerman. Randomness-optimal oblivious sampling. *Random Structures and Algorithms*, 11:345–367, 1997.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de