Uncorrectable Samplable Errors for Low-Rate Codes

Kenji Yasunaga
Kanazawa University
yasunaga@se.kanazawa-u.ac.jp

February 19, 2015

Abstract

We study the correctability of efficiently samplable errors. Specifically, we consider the setting in which errors are efficiently samplable without the knowledge of the code and the transmitted codeword, but the error rate is not bounded. We show that there is an oracle relative to which there exists a samplable distribution of entropy m that is not correctable by efficient coding schemes of rate less than $1 - m/n - \omega(\log n/n)$, where n is the codeword length. We also show that the existence of one-way functions is necessary to derive the impossibility results.

1 Introduction

The theory of error-correcting codes studies the ways of achieving reliable communication over noisy channels. Two of the most studied channel models are probabilistic channels and worst-case channels. In probabilistic channels, errors are considered to be introduced through stochastic processes. A well-studied example is the binary symmetric channel (BSC), in which each bit is independently flipped with some probability $p < 1/2$. In worst-case (or adversarial) channels, errors are introduced adversarially by considering the choice of codes and transmitted codewords under the restriction of the error rate.

In his seminal work [11], Shannon showed that reliable communication can be achieved over BSC if the coding rate is less than $1 - H_2(p)$, where $H_2(\cdot)$ is the binary entropy function and p is the crossover probability of BSC. In contrast, it is known that reliable communication cannot be achieved over worst-case channels when the error rate is at least $1/4$ unless the coding rate tends to zero [10].

As intermediate channels between these two channels, Lipton [8] introduced *computationally-bounded channels*, where errors are introduced by polynomial-time computation. He showed that reliable communication can be achieved at the coding rate less than $1 - H_2(p)$ in the shared randomness setting, where $p < 1$ is the error rate. Micali et al. [9] present reliable coding schemes in the public-key infrastructure setting. Guruswami and Smith [6] showed reliable coding schemes without assuming the shared randomness or the public-key infrastructure. Note that these work [8, 9, 6] consider the settings in which channels are computationally-bounded and the error rate is bounded.

In this work, we focus on computationally-bounded channels with *unbounded* error rate. We do not assume the shared randomness or the public-key infrastructure. The error-correction problem

in such a setting was studied in [16]. The paper [16] showed several results on the correctability in *samplable-additive channels*, where errors are sampled by polynomial-time computation without the knowledge of transmitted codewords. Samplable-additive channels are relatively simple channel models since error distributions of these channels are identical for every coding scheme and transmitted codeword. BSC is an example of samplable-additive channels. Studying the correctability of samplable-additive channels can reveal what computational structure of errors can help to achieve error correction.

As a positive result, it is shown in [16] that if the error vectors form a linear subspace, there is an explicit optimal-rate coding scheme that corrects these errors. A negative result of [16] is that it seems difficult to correct errors only by their samplability even when the entropy of errors is logarithmically small. Specifically, it is shown that there is an oracle relative to which there exists a samplable distribution Z with Shannon entropy $H(Z) = \omega(\log n)$ that is not correctable by coding schemes of rate $\omega(\log n/n)$ that employ *efficient syndrome decoding*, where n is the codeword length. This result implies the impossibility of correcting every samplable errors with low entropy by a black-box way. However, this implication is quite restrictive since the syndrome decoding can be employed only for linear codes, and there are many efficient decoding algorithms other than syndrome decoding in the coding theory literature. Furthermore, the general syndrome-decoding problem is known to be NP-hard [1].

1.1 Our Contributions

Uncorrectable Errors for Low-Rate Codes. In this work, we show that there is an oracle relative to which there exists a samplable distribution Z with $H(Z) = m$ that is not correctable by *efficient coding schemes* of rate $R < 1 - m/n - \omega(\log n/n)$. Namely, we could remove the restriction of syndrome decoding, but the range of rates for the negative result is different. Our negative result implies the impossibility of correcting every samplable errors by low-rate coding schemes in a black-box way.

To derive this result, we use the technique of Wee [15], which is based on the reconstruction paradigm of Gennaro and Trevisan [4]. Wee [15] showed that there is an oracle relative to which there is a samplable distribution of entropy $\omega(\log n)$ that cannot be compressed to length less than $n - \omega(\log n)$ by efficient compression. We use his technique for the problem of error correction. Specifically, we show that if a samplable distribution with a sampler S is efficiently correctable, then the function of S has a short description, and thus efficient coding schemes cannot correct every samplable distribution.

Our negative result seems counterintuitive, since it is, in general, easier to construct low-rate codes than high-rate codes, but the result implies the impossibility of constructing low-rate codes. The reason for such a result is that a low-rate code has a short description, and thus the correctability by such a code helps describe the samplable errors shortly. In the result of [16], in which only linear codes with syndrome decoding are treated, a high-rate code can have a short description of the errors. This is because, in linear codes with syndrome decoding, each correctable error has a one-to-one correspondence with a syndrome, which can be described using $n(1 - R)$ bits for codes with rate R.

Necessity of One-Way Functions. It is observed in [16] that if an error distribution is pseudorandom, it is impossible to correct errors by efficient coding schemes. This negative result implies that, assuming the existence of one-way functions, there is a samplable distribution of entropy n^ϵ

Table 1: Correctability of Samplable Additive-Error Z

$H(Z)$	**Correctabilities**	**Assumptions**	**References**
0	Efficiently correctable for any R	No	Trivial
$\omega(\log n)$	$\exists$ Z not correctable for $R > \omega(\frac{\log n}{n})$ by efficient syndrome decoding	Oracle access	[16]
$\omega(\log n)$	$\exists$ Z not efficiently correctable for $R < 1 - \frac{m}{n} - \omega\left(\frac{\log n}{n}\right)$	Oracle access	This work (Corollary 1)
n^ϵ $(0 < \epsilon < 1)$	$\exists Z$ not efficiently correctable for any R	OWF	[16]
$0 \leq m \leq n$	$\forall$ flat Z is efficiently correctable for $R \leq 1 - \frac{m}{n} - O(\frac{\log n}{n})$	No OWF	This work (Theorem 3)
$0 \leq m \leq n$	$\forall$ linear subspace Z of dimension m is correctable for $R \leq 1 - \frac{m}{n}$	No	[16]
$0 \leq m \leq n$	$\forall$ flat Z is not correctable for $R > 1 - \frac{m}{n} + O(\frac{1}{n})$	No	[16]
$nH_2(p)$ $(0 < p < 1)$	$\exists Z$ not correctable for $R > 1 - H_2(p)$	No	Capacity of BSC
n	Not correctable for any R	No	Trivial

for any $0 < \epsilon < 1$ that is not correctable by polynomial time. It remained open to prove this result without assuming the existence of one-way functions. We show that the existence of one-way functions is necessary to derive impossibility results. Specifically, we show that if one-way functions do not exist, then any samplable flat distribution of entropy m is correctable by an efficient coding scheme of rate $1 - m/n - O(\log n/n)$.

The results of this work and [16] are summarized in Table 1, where R denotes the rate of coding schemes.

1.2 Related Work

The notion of computationally-bounded channel was introduced by Lipton [8]. He showed that the Shannon capacity can be achieved in the shared randomness setting. Micali et al. [9] considered a similar channel model in a public-key setting. Guruswami and Smith [6] gave constructions of optimal-rate codes for worst-case additive channels and time/space-bounded channels. In their setting, the error rate is bounded by some constant $p < 1$. They also gave impossibility results of unique decoding when $p \geq 1/4$, but their results can be applied to channels that use the information on the code and the transmitted codeword. In this work, we give an impossibility result even for channels that do no use such information.

Samplable distributions have been also studied in the context of data compression [5, 13, 15], randomness extractor [12, 14, 2], and randomness condenser [3].

2 Preliminaries

For a distribution X, we write $x \sim X$ to indicate that x is chosen according to X. We may use X also as a random variable distributed according to X. The *support* of X is $\mathrm{Supp}(X) = \{x : \Pr_X(x) \neq 0\}$, where $\Pr_X(x)$ is the probability that X assigns to x. The *Shannon entropy* of X is $H(X) = E_{x\sim X}[-\log \Pr_X(x)]$. A *flat distribution* is a distribution that is uniform over its support. For $n \in \mathbb{N}$, we write U_n as the uniform distribution over $\{0,1\}^n$.

We define the notion of additive-error correcting codes.

Definition 1. *For two functions $\mathsf{Enc} : \mathbb{F}^k \to \mathbb{F}^n$ and $\mathsf{Dec} : \mathbb{F}^n \to \mathbb{F}^k$, and a distribution Z over $\mathbb{F}^n$, where $\mathbb{F}$ is a finite field, we say $(\mathsf{Enc}, \mathsf{Dec})$* corrects (additive error) *Z with error ϵ if for any $x \in \mathbb{F}^k$, we have that $\Pr_{z\sim Z_n}[\mathsf{Dec}(\mathsf{Enc}(x) + z) \neq x] \leq \epsilon$. The* rate *of $(\mathsf{Enc}, \mathsf{Dec})$ is k/n.*

Definition 2. *A distribution Z is said to be* correctable with rate R and error ϵ *if there is a pair of functions $(\mathsf{Enc}, \mathsf{Dec})$ of rate R that corrects Z with error ϵ.*

We call a pair $(\mathsf{Enc}, \mathsf{Dec})$ a *coding scheme* or simply *code*. The coding scheme is called *efficient* if Enc and Dec can be computed in polynomial-time in n. The code is called *linear* if Enc is a linear mapping, that is, for any $x, y \in \mathbb{F}^n$ and $a, b \in \mathbb{F}$, $\mathsf{Enc}(ax + by) = a\mathsf{Enc}(x) + b\mathsf{Enc}(y)$. If $|\mathbb{F}| = 2$, we may use $\{0,1\}$ instead of $\mathbb{F}$.

Next, we define syndrome decoding for linear codes.

Definition 3. *For a linear code $(\mathsf{Enc}, \mathsf{Dec})$, Dec is said to be a* syndrome decoding *if there is a function Rec such that $\mathsf{Dec}(y) = (y - \mathsf{Rec}(y \cdot H^\perp)) \cdot G^{-1}$, where $G \in \mathbb{F}^{Rn\times n}$ satisfies that $\mathsf{Enc}(x) = x \cdot G$ for $x \in \mathbb{F}^{Rn}$, and $H \in \mathbb{F}^{n\times Rn}$ is a dual matrix for G (i.e., $GH^\perp = 0$).*

Finally, we introduce the notion of samplable distributions.

Definition 4. *A distribution family $Z = \{Z_n\}_{n\in\mathbb{N}}$ is said to be* samplable *if there is a probabilistic polynomial-time algorithm S such that $S(1^n)$ is distributed according to Z_n for every $n \in \mathbb{N}$.*

3 Uncorrectable Errors for Low-Rate Codes

We show that there is an oracle relative to which there exists a samplable distribution that is not correctable by efficient coding schemes with low rate.

Let $N = 2^n, K = 2^k, M = 2^m$. Let $\mathcal{F}$ be the set of injective functions $f : \{0,1\}^m \to \{0,1\}^n$. For each $f \in \mathcal{F}$, define an oracle $\mathcal{O}_f$ such that

$$\mathcal{O}_f(b, y) = \begin{cases} \mathcal{O}_f^S(y) & \text{if } b = 0 \\ \mathcal{O}_f^M(y) & \text{if } b = 1 \end{cases}, \quad \mathcal{O}_f^M(y) = \begin{cases} 1 & \text{if } y \in f(\{0,1\}^m) \\ 0 & \text{if } y \notin f(\{0,1\}^m) \end{cases}, \quad \mathcal{O}_f^S(y) = f(y).$$

Let $\mathsf{correctf}$ be the set of functions $f \in \mathcal{F}$ for which there exist oracle circuits $(\mathsf{Enc}, \mathsf{Dec})$ that make q queries to oracle $\mathcal{O}_f$ and correct $f(U_m)$ with rate k/n. For each $f \in \mathcal{F}$ and the corresponding $(\mathsf{Enc}, \mathsf{Dec})$, we define

$$\begin{aligned} \mathsf{invert}_f = \{y \in \{0,1\}^m : &\text{for any } x \in \{0,1\}^k, \text{ on input } \mathsf{Enc}(x) + f(y), \\ &\mathsf{Dec} \text{ queries } \mathcal{O}_f^S \text{ on } y\}, \\ \mathsf{forge}_f = \{y \in \{0,1\}^m : &\text{for some } x \in \{0,1\}^k, \text{ on input } \mathsf{Enc}(x) + f(y), \\ &\mathsf{Dec} \text{ does not query } \mathcal{O}_f^S \text{ on } y\}. \end{aligned}$$

Note that invert_f and forge_f is a partition of $\{0,1\}^m$. We also define

$$\mathsf{invertible} = \{f \in \mathsf{correctf} : |\mathsf{invert}_f| > \epsilon \cdot 2^m\},$$
$$\mathsf{forgeable} = \{f \in \mathsf{correctf} : |\mathsf{forge}_f| \geq \delta \cdot 2^m\},$$

where ϵ and δ are any constants satisfying $\epsilon + \delta = 1$. Note that $\mathsf{correctf} = \mathsf{invertible} \cup \mathsf{forgeable}$.

Intuitively, if f is in $\mathsf{invertible}$, then there is a small circuit that inverts f. This is done by computing $\mathsf{Enc}(x) + f(y)$ and monitoring oracle queries that $\mathsf{Dec}(\mathsf{Enc}(x) + f(y))$ makes to $\mathcal{O}_f^S$. Since a random function is one-way with high probability, we can show that the size of invertible functions, i.e., $\mathsf{invertible}$, is small. Similarly, if f is in $\mathsf{forgeable}$, then Dec corrects $f(y)$ without querying $\mathcal{O}_f^S$ on y. This means that $f(y)$ can be described using Dec and $\mathsf{Enc}(x) + f(y)$, and thus if $\mathsf{Enc}(x) + f(y)$ has a short description, the size of $\mathsf{forgeable}$ is small.

To argue the above intuition formally, we use the reconstruction paradigm of [4]. Then, we show that both $\mathsf{invertible}$ and $\mathsf{forgeable}$ are small.

First, we show that $f \in \mathsf{invertible}$ has a short description.

Lemma 1. *Take any $f \in \mathsf{invertible}$ and the corresponding oracle circuit $(\mathsf{Enc}, \mathsf{Dec})$ that makes q queries to $\mathcal{O}_f$ and corrects $f(U_m)$ with rate k/n. Then f can be described using at most*

$$\log \binom{N}{c} + \log \binom{M}{c} + \log \left(\binom{N-c}{M-c} (M-c)! \right)$$

bits, given $(\mathsf{Enc}, \mathsf{Dec})$, where $c = \epsilon M/q$.

Proof. First, consider an oracle circuit A such that, on input z, A picks any $x \in \{0,1\}^k$ and simulates Dec on input $\mathsf{Enc}(x) + z$. Then, for any $y \in \mathsf{invert}_f$, on input $f(y)$, A outputs y by making at most q queries to $\mathcal{O}_f$.

Next, we show that for any $f \in \mathsf{invertible}$, f has a short description given A. Without loss of generality, we assume that A makes distinct queries to $\mathcal{O}_f^S$. We also assume that on input $f(y)$, A always queries $\mathcal{O}_f^S$ on y before it outputs y. We will show that there is a subset $T \subseteq f(\mathsf{invert}_f)$ such that f can be described given T, $B(T)$, $f|_{\{0,1\}^m \setminus B(T)}$, where $B(T) = \{y \in \{0,1\}^m : y \leftarrow A(z), z \in T\}$.

We describe how to construct T below.

CONSTRUCT-T:

1. Initially, T is empty, and all elements in $T^* = f(\mathsf{invert}_f)$ are candidates for inclusion in T.
2. Choose the lexicographically smallest z from T^*, put z in T, and remove z from T^*.
3. Simulate A on input z, and halt the simulation immediately after A queries $\mathcal{O}_f^S$ on y. Let $y'_1, \ldots, y'_p$ be the queries that A makes to $\mathcal{O}_f^S$, where $y'_p = y$ and $p \leq q$.
   - Remove $f(y'_1), \ldots, f(y'_{p-1})$ from T^*. (This means that these elements will never belong to T, and in simulating $A(z)$ in the recovering phase, the answers to these queries are made by using the look-up table for f.)
   - Continue to remove the lexicographically smallest z from T^* until we have removed exactly $q-1$ elements in Step 3.
4. Return to Step 2.

Next, we describe how to reconstruct f from T, $B(T)$, and $f|_{\{0,1\}^m \setminus B(T)}$. We show how to recover the look-up table for f on values in $B(T)$.

RECOVER-f:

1. Choose the lexicographically smallest element $z \in T$, and remove it from T.
2. Simulate A on input z, and halt the simulation immediately after A queries $\mathcal{O}_f^S$ on y for which the answer does not exist in the look-up table for f. Since the query y satisfies that $y = f^{-1}(z)$, add the entry (y, z) to the look-up table.
   In what follows, we explain why we can correctly simulate $A(z)$.
   - Since $B(T)$ and $f|_{\{0,1\}^m \setminus B(T)}$ are given, we can answer all queries to O_f^M.
   - For any query y' to O_f^S, it must be either (1) $y' \notin B(T)$, or (2) y' is the output of A on input z' such that $z' \in W$ and z' is lexicographically smaller than z. In either case, the look-up table has the corresponding entry, and thus we can answer the query.
3. Return to Step 1.

In each iteration in Construct-T, we add one element to T and remove exactly q element from T^*. Since initially the size of $T^* = f(\mathsf{invert}_f)$ is ϵM, the size of T in the end is $c = \epsilon M/q$.

The sets T and $B(T)$, and the look-up table for $f|_{\{0,1\}^m \setminus B(T)}$ can be described using $\log \binom{N}{c}$, $\log \binom{M}{c}$, and $\log(\binom{N-c}{M-c}(M-c)!)$, respectively. Therefore, the statement follows. □

We show that the fraction of $f \in \mathcal{F}$ for which $f \in \mathsf{invertible}$ and $f(U_m)$ is correctable is small.

Lemma 2. *Let $(\mathsf{Enc}, \mathsf{Dec})$ be oracle circuits of size s. If $m > 3\log s + \log n + O(1)$, then the fraction of functions $f \in \mathcal{F}$ such that $f \in \mathsf{invertible}$ and $(\mathsf{Enc}, \mathsf{Dec})$ corrects $f(U_m)$ is less than $2^{-(sn\log s+1)}$ for all sufficiently large n.*

Proof. It follows from Lemma 1 that, given $(\mathsf{Enc}, \mathsf{Dec})$, the fraction is

$$\frac{|\mathsf{invertible}|}{\binom{N}{M}M!} \leq \frac{\binom{N}{c}\binom{M}{c}\binom{N-c}{M-c}(M-c)!}{\binom{N}{M}M!} = \frac{\binom{M}{c}}{c!},$$

where $c = \epsilon M/(qK)$. By using the fact that $q \leq s$ and the inequalities $\binom{n}{k} < \left(\frac{en}{k}\right)^k$ and $n! > \left(\frac{n}{e}\right)^n$, the expression is upper bounded by

$$\left(\frac{eM}{c}\right)^c \left(\frac{e}{c}\right)^c = \left(\frac{e^2q^2}{\epsilon^2 M}\right)^{\epsilon M/q} < \left(\frac{1}{2}\right)^{ns\log s+1}$$

for all sufficiently large n. □

Next, we show that $\mathsf{forgeable}$ has a short description.

Lemma 3. *Take any $f \in \mathsf{forgeable}$ and the corresponding oracle circuit $(\mathsf{Enc}, \mathsf{Dec})$ that makes q queries to $\mathcal{O}_f$ and corrects $f(U_m)$ with rate k/n. Then f can be described using at most*

$$\log \binom{M}{d} + \log\left(\binom{N-d}{M-d}(M-d)!\right) + d(k+m+\log q)$$

bits, given $(\mathsf{Enc}, \mathsf{Dec})$, where $d = \delta M/q$.

Proof. First, consider an oracle circuit A such that, on input w, A obtains x by simulating Dec on input w, queries $\mathcal{O}_f^M$ on $w - \mathsf{Enc}(x)$, and outputs $\perp$ if $\mathcal{O}_f^M(w - \mathsf{Enc}(x)) = 0$, and x otherwise. Then, A satisfies that, on input w, A outputs $\perp$ if $w \notin \mathsf{Enc}(\{0,1\}^k) + f(\{0,1\}^m)$, and $\mathsf{Dec}(w)$ otherwise.

Next, we show that for any $f \in \mathsf{forgeable}$, f has a short description given A. Without loss of generality, we assume that A makes distinct queries to $\mathcal{O}_f^S$ and $\mathcal{O}_f^M$. We also assume that for $x \in \{0,1\}^k$ and $y \in \{0,1\}^m$, $A(\mathsf{Enc}(x) + f(y))$ always queries $\mathcal{O}_f^M$ on $f(y)$ before it outputs x. Note that for $y \in \mathsf{forge}_f$, there is some $x \in \{0,1\}^k$ such that, on input $\mathsf{Enc}(x) + f(y)$, A does not query $\mathcal{O}_f^S$ on y.

We will show that there is a subset $Y \subseteq \mathsf{forge}_f$ such that f can be described given Y, $f|_{\{0,1\}^m \setminus Y}$, and $\{(x_y, a_y, b_y) \in \{0,1\}^k \times [M] \times [q] : y \in Y\}$ of a set of advice strings. For $x \in \{0,1\}^k$, we define $D(x) = \{\mathsf{Enc}(x) + f(y) : y \in \{0,1\}^m\}$. Note that $|D(x)| = M$ for any $x \in \{0,1\}^k$.

We describe how to construct Y below.

Construct-Y:

1. Initially, Y is empty. All elements in $Y^* = \mathsf{forge}_f$ are candidates for inclusion in Y. For every $x \in \{0,1\}^k$, set $D_x = \{\mathsf{Enc}(x) + f(y) : y \in \mathsf{forge}_f\}$. We write $\mathcal{D}_k = \bigcup_{x \in \{0,1\}^k} D_x$.
2. Choose the lexicographically smallest y from Y^*, put y in Y, and remove y from Y^*.
3. Choose the lexicographically smallest w from the set of $\mathsf{Enc}(x) + f(y) \in D_x$ such that A does not query $\mathcal{O}_f^S$ on y. If $w = \mathsf{Enc}(x) + f(y)$, set $x_y = x$. Then, for every $x' \in \{0,1\}^k$, remove $\mathsf{Enc}(x') + f(y)$ from $D_{x'}$. (This removal means that hereafter there are no elements in $\mathcal{D}_k$ for which A outputs some x such that $f(y)$ is the error vector.) When w is the lexicographically t-th smallest element in $D(x)$, set $a_y = t$ (so that we can recognize that the a_y-th element in $D(x)$ is w in the recovering phase).
4. Simulate A on input w, and halt the simulation immediately after A queries $\mathcal{O}_f^M$ on $f(y)$. Let $y'_1, \ldots, y'_p$ be the queries that A makes to $\mathcal{O}_f^S$, and $z'_1, \ldots, z'_r = f(y)$ be the queries that A makes to $\mathcal{O}_f^M$. Set $b_y = r$ (so that we can recognize that the b_y-th query that Dec makes to $\mathcal{O}_f^M$ is $f(y)$ in the recovering phase).
   (a) For every $x' \in \{0,1\}^k$, remove $\mathsf{Enc}(x') + f(y'_1), \ldots, \mathsf{Enc}(x') + f(y'_p)$ from $D_{x'}$.
   (b) For every $i \in [p]$, if $z'_i \in f(\mathsf{forge}_f)$, then for every $x' \in \{0,1\}^k$, remove $\mathsf{Enc}(x') + z'_i$ from $D_{x'}$, and otherwise, do nothing.
   (c) Continue to remove the elements $\mathsf{Enc}(x') + f(y)$ from $D_{x'}$ for every $x' \in \{0,1\}^k$ for the lexicographically smallest $w = \mathsf{Enc}(x) + f(y) \in \mathcal{D}_k$ until we have removed exactly $(q-1)K$ elements from $\mathcal{D}_k$ in Step 4.
5. Return to Step 2.

Next, we describe how to construct f from Y, $f|_{\{0,1\}^m \setminus Y}$, and $\{(x_y, a_y, b_y) \in \{0,1\}^k \times [M] \times [q] : y \in Y\}$. We show how to recover the look-up table for f on values in Y.

Recover-f:

1. Choose the lexicographically smallest $y \in Y$, and remove it from Y. Then, choose the lexicographically a_y-th smallest element w from $D(x_y)$.

2. Simulate A on input w, and halt the simulation immediately after A makes the b_y-th query to $\mathcal{O}_f^M$. Since the b_y-th query is $f(y)$, add the entry $(y, f(y))$ to the look-up table.
   In what follows, we explain why we can correctly simulate $A(w)$.
   - For any query y' to $\mathcal{O}_f^S$, it must be either (1) $y' \notin Y$ or (2) y' is lexicographically smaller than y. In case (1), we can answer the query by using $f|_{\{0,1\}^m \setminus Y}$. In case (2), since y was chosen as the lexicographically smallest element such that A does not query $\mathcal{O}_f^S$ on y, the look-up table has the answer to the query.
   - Consider any of the first $b_y - 1$ queries z' to $\mathcal{O}_f^M$. If $z' \in f(\{0,1\}^m)$, namely $z' = f(y')$ for some y', then it must be either (1) $y' \notin Y$ or (2) y' is lexicographically smaller than y. In either case, the look-up table has the entry (y', z'). If $z' \notin f(\{0,1\}^m)$, there is no entry for z' in the look-up table. Thus, we can answer the query by saying "yes" if z' is in the look-up table, and "no" otherwise.
3. Return to Step 1.

In each iteration in CONSTRUCT-Y, we add one element to Y and remove exactly qK elements from $\mathcal{D}_k$. Since initially the size of $\mathcal{D}_k$ is at least δKM, the size of Y in the end is at least $d = \delta M/q$.

The set Y, the look-up table for $f|_{\{0,1\}^m \setminus Y}$, the sets $\{(x_y, a_y, b_y) \in \{0,1\}^k \times [M] \times [q] : y \in Y\}$ can be described using $\binom{M}{d}$, $\log(\binom{N-d}{M-d}(M-d)!)$, and $d(k+m+\log q)$ bits respectively. Therefore, the statement follows. □

We show that the fraction of $f \in \mathcal{F}$ for which $f \in \mathsf{forgeable}$ and $f(U_m)$ is correctable is small.

Lemma 4. *Let $(\mathsf{Enc}, \mathsf{Dec})$ be oracle circuits of size s. If $m > 3\log s + \log n + O(1)$ and $m < n - k - 2\log s - O(1)$, then the fraction of functions $f \in \mathcal{F}$ such that $f \in \mathsf{forgeable}$ and $(\mathsf{Enc}, \mathsf{Dec})$ corrects $f(U_m)$ is less than $2^{-(sn\log s+1)}$ for all sufficiently large n.*

Proof. It follows from Lemma 3 that, given $(\mathsf{Enc}, \mathsf{Dec})$, the fraction is

$$\frac{|\mathsf{forgeable}|}{\binom{N}{M}M!} \le \frac{\binom{M}{d}\binom{N-d}{M-d}(M-d)!}{\binom{N}{M}M!} 2^{d(k+m+\log q)} = \frac{\binom{M}{d}}{\binom{N}{d}d!}(qKM)^d,$$

where $d = \delta M/q$. By using the fact that $q \le s$ and the inequalities $\binom{n}{k} < \left(\frac{en}{k}\right)^k$, $\binom{n}{k} > \left(\frac{n}{k}\right)^k$, and $n! > \left(\frac{n}{e}\right)^n$, the expression is upper bounded by

$$\left(\frac{eM}{d}\right)^d \left(\frac{d}{N}\right)^d \left(\frac{e}{d}\right)^d (qKM)^d = \left(\frac{e^2q^2KM}{\delta N}\right)^{\delta M/q} < \left(\frac{1}{2}\right)^{ns\log s+1}$$

for all sufficiently large n. □

We obtain our main result.

Theorem 1. *For any m and k satisfying $3\log s + \log n + O(1) < m < n - k - 2\log s - O(1)$, there exist injective functions $f : \{0,1\}^m \to \{0,1\}^n$ such that, given oracle access to $\mathcal{O}_f$, (1) $f(U_m)$ is samplable and has entropy m, and (2) $f(U_m)$ cannot be corrected with rate k/n by oracle circuits of size s.*

Proof. Since $\mathsf{correctf} = \mathsf{invertible} \cup \mathsf{forgeable}$, it follows from Lemmas 2 and 4 that for a fixed $(\mathsf{Enc}, \mathsf{Dec})$ of size s, the fraction of functions $f \in \mathcal{F}$ such that $(\mathsf{Enc}, \mathsf{Dec})$ corrects $f(U_m)$ with rate k/n is less than $2^{-(sn \log s)}$. Since there are at most $2^{sn \log s}$ circuits of size s, there are functions $f \in \mathcal{F}$ such that $f(U_m)$ cannot be corrected with rate k/n by oracle circuits of size s. Given oracle access to $\mathcal{O}_f$, $f(U_m)$ is samplable. Since f is injective, $f(U_m)$ has entropy m. □

The following corollary immediately follows.

Corollary 1. *For any m and k satisfying $\omega(\log n) < m < n - k - \omega(\log n)$, there exists an oracle relative to which there exists a samplable distribution of entropy m that cannot be corrected with rate k/n by polynomial size circuits.*

4 Necessity of One-Way Functions

In this section, we show that if one-way functions do not exist, then any samplable flat distribution of entropy m is correctable by an efficient coding scheme of rate $1 - m/n - O(\log n/n)$. For this, we use a technique used in the proof of [15, Theorem 6.3] that shows the necessity of one-way functions for separating pseudoentropy and compressibility. We observe that in its proof, a family of linear hash functions is used for giving an efficient compression function. Since a linear compression function is a dual object of a linear code that corrects additive errors, we can use a family of linear hash functions for constructing an efficient decoder.

Definition 5 ([7])**.** *We say a function f is* distributionally one-way *if it is computable in polynomial time and there exists a constant $c > 0$ such that for every probabilistic polynomial-time algorithm A, the statistical distance between $(x, f(x))$ and $(A(f(x)), f(x))$ is at least $1/n^c$, where $x \sim U_n$.*

Theorem 2 ([7])**.** *If there is a distributionally one-way function, then there is a one-way function.*

Theorem 3. *If one-way functions do not exist, then any samplable flat distribution Z over $\{0,1\}^n$ of entropy m can be corrected with rate $1 - m/n - (c \log n)/n$ and error $O(n^{-c})$ for any constant $c > 0$ by polynomial-time coding schemes.*

Proof. Let $Z = f(U_r)$ for an efficiently computable function f. Consider a family of linear universal hash functions $\mathcal{H} = \{h : \{0,1\}^n \to \{0,1\}^{n+2c\log n}\}$, where the universality means that for any distinct $x, y \in \{0,1\}^n$, $\Pr_{h\in\mathcal{H}}[h(x) = h(y)] \leq 2^{m+2c\log n}$, and the linearity means that for any $x, y \in \{0,1\}^n$ and $a, b \in \{0,1\}$, $h(ax + by) = ah(x) + bh(y)$. For each $h \in \mathcal{H}$, we define $C_h = \{x \in \mathrm{Supp}(Z) : \exists y \in \mathrm{Supp}(Z) \text{ s.t. } y \neq x \wedge h(x) = h(y)\}$. Namely, C_h is the set of inputs with collisions under h. By a union bound, it holds that for any $x \in \mathrm{Supp}(Z)$,

$$\Pr_{h\in\mathcal{H}}[\exists y \in \mathrm{Supp}(Z) : y \neq x \wedge h(y) = h(x)] \leq \frac{2^m}{2^{m+2c\log n}} = \frac{1}{n^{2c}}.$$

Thus, $E[|C_h|] \leq 2^m/n^{2c}$. We say $h \in \mathcal{H}$ is good if $|C_h| \leq 2^m/n^c$. By Markov's inequality, we have that $\Pr_{h\in\mathcal{H}}[|C_h| > 2^m/n^c] < 1/n^c$.

Consider the function $g : \{0,1\}^n \times \mathcal{H} \to \mathcal{H} \times \{0,1\}^{m+2\log n}$ given by $g(y, h) = (h, h(f(y)))$. Note that g is polynomial-time computable. By the assumption that one-way functions do not exist, and thus distributionally one-way functions do not exist, there is a polynomial-time algorithm A such

that the statistical distance between $(y, h, g(y,h))$ and $(A(g(y,h)), g(y,h))$ is at most n^{-c} for any constant $c > 0$, where $y \sim U_r$ and $h \in \mathcal{H}$. Then, it holds that

$$\Pr_{A,y,h}[g(A(g(y,h))) = g(y,h)] \geq 1 - \frac{1}{n^c},$$

where the probability is taken over the coins of A, $y \sim U_r$, and $h \in \mathcal{H}$. Thus, we have that

$$\Pr_{A,y,h}[g(A(g(y,h))) = g(y,h) \wedge h \text{ is good}] \geq 1 - \frac{2}{n^c}.$$

By fixing the coins of A and $h \in \mathcal{H}$, it holds that there are deterministic algorithm A' and $h_0 \in \mathcal{H}$ such that h_0 is good and

$$\Pr_{y}[g(A'(g(y,h_0))) = g(y,h_0)] \geq 1 - \frac{2}{n^c}.$$

For $y \in \{0,1\}^r$ satisfying $g(A'(g(y,h_0))) = g(y,h_0)$, we write $A'(g(y,h_0)) = (y', h')$, where $A'_1(g(y,h_0)) = y'$ and $A'_2(g(y,h_0)) = h'$. Then, it holds that $h' = h_0$ and $h_0(f(y)) = h_0(f(y'))$. Furthermore, since h_0 is good, $\Pr_y[f(y) \notin C_{h_0}] \geq 1 - 1/n^c$. Let $H_0 \in \{0,1\}^{(m+c\log n)\times n}$ be a matrix such that $xH_0^T = h_0(x)$ for $x \in \{0,1\}^n$. (Such matrices exist since $\mathcal{H}$ is a set of linear hash functions.) Consider a linear coding scheme in which H_0 is employed as the parity check matrix, and A'_1 is employed for recovering errors from syndromes. That is, $\mathsf{Enc}(x) = xG$ for a matrix $G \in \{0,1\}^{(n-m-c\log n)\times n}$ satisfying $GH_0^T = 0$, and $\mathsf{Dec}(y) = (y - f(A'_1(h_0, yH_0^T)))G^{-1}$. Then, for any $x \in \{0,1\}^m$,

$$\begin{aligned}
&\Pr_{y\sim U_r}[\mathsf{Dec}(\mathsf{Enc}(x) + f(y)) = x] \\
&= \Pr_{y\sim U_r}[\mathsf{Enc}(x) + f(y) - f(A'_1(h_0, (\mathsf{Enc}(x) + f(y))H_0^T)) = xG] \\
&= \Pr_{y\sim U_r}[f(A'_1(g(y,h_0))) = f(y)],
\end{aligned}$$

where we use the property that $\mathsf{Enc}(x) = xG$, $GH_0^T = 0$, and $xH_0^T = h_0(x)$. Since the probability that $g(A_0(g(y,h_0))) = g(y,h_0)$ is at least $1 - 2/n^c$, and for any $y \in \{0,1\}^r$ satisfying $g(A_0(g(y,h_0))) = g(y,h_0)$, $\Pr_y[f(y) \notin C_{h_0}] \geq 1 - 1/n^c$, we have that

$$\Pr_{y\sim U_r}[f(A'_1(g(y,h_0))) = f(y)] \geq 1 - \frac{3}{n^c}.$$

Hence the statement follows. □

Acknowledgments

This work was supported in part by JSPS Grant-in-Aid for Scientific Research Numbers 23500010, 23700010, 24240001, and 25106509.

References

[1] E. R. Berlekamp, R. J. McEliece, and H. C. A. van Tilborg. On the inherent intractability of certain coding problems (corresp.). *IEEE Transactions on Information Theory*, 24(3):384–386, 1978.

[2] A. De and T. Watson. Extractors and lower bounds for locally samplable sources. *TOCT*, 4(1):3, 2012.

[3] Y. Dodis, T. Ristenpart, and S. P. Vadhan. Randomness condensers for efficiently samplable, seed-dependent sources. In R. Cramer, editor, *TCC*, volume 7194 of *Lecture Notes in Computer Science*, pages 618–635. Springer, 2012.

[4] R. Gennaro and L. Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *41st Annual Symposium on Foundations of Computer Science, FOCS 2000.*

[5] A. V. Goldberg and M. Sipser. Compression and ranking. *SIAM J. Comput.*, 20(3):524–536, 1991.

[6] V. Guruswami and A. Smith. Codes for computationally simple channels: Explicit constructions with optimal rate. In *FOCS*, pages 723–732. IEEE Computer Society, 2010.

[7] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *FOCS*, pages 230–235. IEEE Computer Society, 1989.

[8] R. J. Lipton. A new approach to information theory. In P. Enjalbert, E. W. Mayr, and K. W. Wagner, editors, *STACS*, volume 775 of *Lecture Notes in Computer Science*, pages 699–708. Springer, 1994.

[9] S. Micali, C. Peikert, M. Sudan, and D. A. Wilson. Optimal error correction for computationally bounded noise. *IEEE Transactions on Information Theory*, 56(11):5673–5680, 2010.

[10] M. Plotkin. Binary codes with specified minimum distance. *IRE Transactions on Information Theory*, 6(4):445–450, 1960.

[11] C. E. Shannon. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423,623–656, 1948.

[12] L. Trevisan and S. P. Vadhan. Extracting randomness from samplable distributions. In *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, 12-14 November 2000, Redondo Beach, California, USA*, pages 32–42. IEEE Computer Society, 2000.

[13] L. Trevisan, S. P. Vadhan, and D. Zuckerman. Compression of samplable sources. *Computational Complexity*, 14(3):186–227, 2005.

[14] E. Viola. Extractors for circuit sources. In R. Ostrovsky, editor, *FOCS*, pages 220–229. IEEE, 2011.

[15] H. Wee. On pseudoentropy versus compressibility. In *IEEE Conference on Computational Complexity*, pages 29–41. IEEE Computer Society, 2004.

[16] K. Yasunaga. Correction of samplable additive errors. In *2014 IEEE International Symposium on Information Theory, Honolulu, HI, USA, June 29 - July 4, 2014*, pages 1066–1070. IEEE, 2014.