DNA-Based Storage: Models and Fundamental Limits

Ilan Shomorony, *Member, IEEE*, and Reinhard Heckel, *Member, IEEE*

***Abstract*—Due to its longevity and enormous information density, DNA is an attractive medium for archival storage. In this work, we study the fundamental limits and trade-offs of DNA-based storage systems by introducing a new channel model, which we call the noisy shuffling-sampling channel. Motivated by current technological constraints on DNA synthesis and sequencing, this model captures three key distinctive aspects of DNA storage systems: (1) the data is written onto many short DNA molecules; (2) the molecules are corrupted by noise during synthesis and sequencing and (3) the data is read by randomly sampling from the DNA pool. We provide capacity results for this channel under specific noise and sampling assumptions and show that, in many scenarios, a simple index-based coding scheme is optimal.**

***Index Terms*—Data storage, DNA storage, channel capacity.**

I. Introduction

DUE to its longevity and enormous information density, and thanks to rapid advances in technologies for writing (synthesis) and reading (sequencing), DNA is on track to become an attractive medium for archival data storage. DNA is a long molecule made up of four nucleotides (Adenine, Cytosine, Guanine, and Thymine) and, for storage purposes, can be viewed as a string over a four-letter alphabet. While in a living cell a DNA molecule can consist of millions of nucleotides, due to technological constraints, it is difficult and inefficient to synthesize long strands of DNA. Thus, in practice, data is stored on short DNA molecules which are preserved in a DNA pool and cannot be spatially ordered.

In recent years, several groups demonstrated working DNA storage systems [1]–[7]. In these systems, information was stored on molecules of no longer than one or two hundred nucleotides. At the time of reading, the information is accessed via state-of-the-art sequencing technologies. This corresponds to (randomly) sampling and reading sequences from the pool of DNA. Sequencing is preceded by several cycles of Polymerase Chain Reaction (PCR) amplification. In each cycle each molecule is replicated by a factor of 1.6-1.8. Thus, the proportions of the sequences in the DNA mixture just before sequencing and the probability that a given sequence is read depends on the synthesis method, the PCR steps, and the decay of DNA during storage. Finally, sequencing and in particular synthesis of DNA may lead to insertions, deletions, and substitutions of nucleotides in individual DNA molecules. We refer to [8] for a detailed discussion of the error sources and probabilities for different experimental setups.

Given these constraints, a mathematical model for a DNA storage channel is as follows. Data is written on M DNA molecules, each of length L. From this multi-set of sequences, N sequences are drawn according to some distribution Q, and are then perturbed by the introduction of individual base errors. A critical element of this model is that by drawing N sequences according to some distribution Q, the order of the sequences is lost.

The decoder's goal is to reconstruct the information from the multi-set of N reads. Note that the decoder has no information about which molecules were sampled, and in general a fraction of the original DNA fragments may never be sampled. Our goal is to study the capacity of this channel under different modeling assumptions on the sampling distribution and the base errors that are introduced.

A. Contributions

In this paper we study the fundamental limits of the DNA storage model outlined above. Our analysis aims to reveal the basic relationships and trade-offs between key design parameters and performance goals such as storage density and reading/writing costs. Throughout, we consider the asymptotic regime where $M \to \infty$. The main parameter of interest is the storage capacity C, defined as the maximum number of bits that can be reliably stored per nucleotide (the total number of nucleotides is ML).

a) Capacity in the case of noise-free sequences: We start with a channel without errors in the individual sequences. Thus, randomness is only introduced through the distribution Q, which describes the number of copies we draw from each input sequence. According to Q, some of the individual sequences might never be drawn and others are drawn many times. Our main result for this channel states that if $\lim_{M\to\infty} \frac{L}{\log M} = \beta > 1$, then

$$C = (1 - q_0)(1 - 1/\beta), \tag{1}$$

where q_0 is the probability that a given sequences is never sampled. Interestingly, our result only depends on the

Manuscript received December 15, 2019; revised October 4, 2020; accepted January 26, 2021. Date of publication February 12, 2021; date of current version May 20, 2021. The work of Ilan Shomorony was supported in part by NSF under Grant CCF-2007597. The work of Reinhard Heckel was supported by the IAS at TUM. *(Corresponding author: Ilan Shomorony.)*

Ilan Shomorony is with the Department of Electrical and Computer Engineering, University of Illinois at Urbana–Champaign, Urbana, IL 61801 USA (e-mail: ilanshom@gmail.com).

Reinhard Heckel is with the Department of Electrical and Computer Engineering, Technical University of Munich, 80333 München, Germany, and also with the Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005 USA (e-mail: reinhard.heckel@tum.de).

Communicated by A. Jiang, Associate Editor for Coding Theory.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIT.2021.3058966.

Digital Object Identifier 10.1109/TIT.2021.3058966

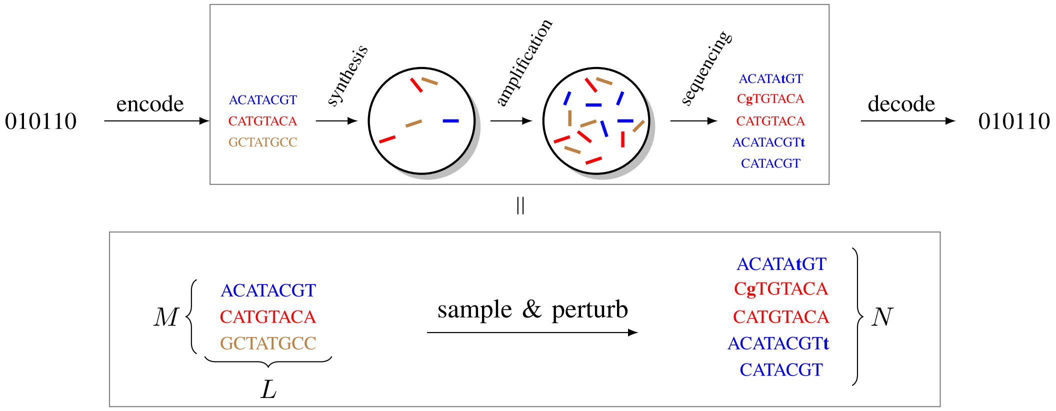

Fig. 1. Channel model for DNA storage systems. The input to the channel is a multi-set of M length-L DNA molecules and the output is a multi-set of N draws from the pool of DNA molecules that are perturbed by insertions, substitutions, and deletions (marked as lowercase and boldface letters).

distribution Q through q_0, which is the probability that a given sequences is never sampled. Moreover, if $\lim_{M\to\infty} \frac{L}{\log M} < 1$, no positive rate is achievable. The factor $1-q_0$ is the loss due to unseen molecules, and $1-1/\beta$ corresponds to the loss due to the unordered fashion of the reading process.

One important implication of our result is that a simple index-based scheme (as commonly used by DNA data storage systems) is optimal; i.e., prefixing each molecule with a unique index incurs no rate loss. More specifically, our result shows that indexing each DNA molecule and employing an erasure code across the molecules is capacity-optimal. Furthermore, the capacity in (1) is only non-trivial if the read length scales as $L = \Theta(\log M)$. For that reason, throughout the paper we focus on the regime $L = \beta \log M$, where β is a positive constant.

Suppose that each sequence is drawn according to a Poisson distribution with mean λ, so that in expectation λM sequences are drawn and λ can be thought of as the sequencing coverage depth. Then, the probability that a sequence is never drawn is $e^{-\lambda}$ and it decays exponentially in the coverage depth. For this scenario, our expression for the capacity suggests that practical systems should not operate at a high coverage depth N/M, as high coverage depth significantly increases the time and cost of reading, but only provides little storage gains. Notice that, in order to guarantee that all M sequences are observed at least once, we need $N = \Omega(M \log M)$ [9], [10]. When M is large, it is wasteful to operate in this regime, as this only gives a marginally larger storage capacity, but the sequencing costs can be exorbitant.

b) Capacity in the case of noisy sequences: Our second contribution is an expression for the capacity for the case where the reading of the sequences is noisy. The goal of this second statement is to capture the effect of errors within sequences, in addition to the shuffling and sampling of the sequences. We assume that the distribution Q with which the sequences are drawn is a simple Bernoulli distribution; i.e., a sequence is either drawn once with probability $1-q_0$ or not drawn with probability q_0. Furthermore we focus on substitution errors within sequences. Thus, we study a noisy shuffling-sampling model where the output sequences are obtained as follows: (i) each original sequence is drawn with probability $1-q$ and not drawn with probability q, (ii) the drawn sequences are shuffled, and (iii) passed through a binary symmetric channel with crossover probability p.

In the low-error regime (where p is sufficiently small), we show that the capacity of this noisy shuffling-sampling channel is given by

$$C = (1-q)(1-H(p)-1/\beta). \tag{2}$$

Note that $1-H(p)$ is the capacity of the binary symmetric channel. As it turns out, (2) can be achieved by treating each length-L sequence as the input to a separate BSC and encoding a unique index into each sequence, and using an erasure outer code to protect against the loss of a q-fraction of the M sequences. For a large set of parameters β and p (described in Section IV), this index-based approach is capacity-optimal. This result provides a theoretical justification for a number of works, starting with [3], which have used a similar coding scheme in real implementations of DNA-based storage systems [3]–[6].

B. Related Literature

Computer scientists and engineers have dreamed of harnessing DNA's storage capabilities already in the 60s [11], [12], and in recent years this idea has developed into an active field of research. In 2012 and 2013 groups lead by Church [1] and Goldman [2] independently stored about a megabyte of data in DNA. In 2015, Grass *et al.* [3] demonstrated that millenia long storage times are possible by protecting the data both physically and information-theoretically, and designed a robust DNA data storage scheme using modern error correcting codes. Later, in the same year, Yazdi *et al.* [4] showed how to selectively access parts of the stored data, and in 2017, Erlich and Zielinski [5] demonstrated that practical DNA storage can achieve very high information densities. In 2018, Organick *et al.* [6] scaled up these techniques and stored about 200 megabytes of data.

The capacity of a DNA storage system under a related model has been studied in an unpublished manuscript by MacKay, Sayer, and Goldman [13]. In their model in the input to the channel consists of a (potentially arbitrarily large) set of DNA molecules of fixed length L, which is not allowed to contain duplicates. The output of the channel are M molecules drawn with replacement from that set. They consider coding over repeated independent storage experiments, and compute the single-letter mutual information over one storage experiment. This indicates that the price of not knowing the ordering of the molecules is logarithmic in the number of synthesized molecules, similar to our main result.

The capacity of a DNA storage system under a different model was studied in [5]. Specifically [5] assumes that each DNA segment is indexed which reduces the channel model to an erasure channel. While this assumption removes the key aspects that we focus on in this paper, namely that DNA molecules are stored in an unordered way and read via random sampling, [5] considers other important constraints, such as homopolymer limitations.

Several recent works have designed coding schemes for DNA storage systems based on this general model, some of which were implemented in proof-of-concept storage systems [1]–[3], [5], [14]. Several papers have studied important additional aspects of the design of a practical DNA storage system. Some of these aspects include DNA synthesis constraints such as sequence composition [4], [5], [15], the asymmetric nature of the DNA sequencing error channel [16], the need for codes that correct insertion errors [17], and the need for techniques to allow random access [4]. The use of fountain codes for DNA storage was considered in [5].

Finally, the recent papers [18] and [19] consider an extension of the channels studied in this paper to the case where each input string can be observed at the output multiple times, with independent noise patterns. We discuss these results in more detail in Section V-B.

II. Problem Setting and Channel Models

An (M, L) DNA storage code $\mathcal{C}$ is a set of codewords, each of which is a list $[x_1^L, \ldots, x_M^L]$ of M strings of length L, together with a decoding procedure. The alphabet Σ is typically $\{\mathsf{A}, \mathsf{C}, \mathsf{G}, \mathsf{T}\}$, corresponding to the four nucleotides that compose DNA. However, to simplify the exposition we focus on the binary case $\Sigma = \{0, 1\}$, and we note that the results can be extended to a general alphabet in a straightforward manner. Throughout the paper we use the word molecule or sequence to refer to each of the stored strings of length L over the alphabet Σ. We study the following general noisy shuffling-sampling channel model:

1) Given that codeword $[x_1^L, \ldots, x_M^L] \in \mathcal{C}$ is chosen, each sequence x_i^L is sampled a number $N_i \sim Q$ of times, for some distribution $Q = (q_0, q_1, \ldots)$, where $q_n = \Pr(N_i = n)$ is the probability that x_i^L is drawn n many times. We let $N = \sum_{i=1}^M N_i$ be the total number of resulting strings, and we define $\lambda := \mathbb{E}[N]/M = \mathbb{E}[N_i]$. The distribution Q models imperfections in synthesis, sequencing, and a loss of whole sequences during storage (see [8] for a detailed discussion on how this distribution looks like for specific choices of sequencing and synthesis technologies).
2) Each of the resulting N strings is passed through a discrete memoryless channel.
3) The resulting N strings are shuffled uniformly at random to yield the output $[y_1^L, \ldots, y_N^L]$. Equivalently, the output of the channel is the (unordered) multi-set of N output sequences $\{y_1^L, \ldots, y_N^L\}$.

A decoding function then maps the received sequences $[y_1^L, \ldots, y_N^L]$ to a message index in $\{1, \ldots, |\mathcal{C}|\}$. The main parameter of interest of a DNA storage system is the storage density, or the storage rate, defined as the number of bits written per DNA base synthesized, i.e.,

$$R := \frac{\log |\mathcal{C}|}{ML}. \tag{3}$$

We consider an asymptotic regime where $M \to \infty$ and we let $L := \beta \log M$ for some fixed β. As our main results show, $L = \Omega(\log M)$ is the asymptotic regime of interest for this problem. We say that the rate R is achievable if there exists a sequence of DNA storage codes $\mathcal{C}_M$ with rate R such that the decoding error probability tends to 0 as $M \to \infty$.

III. Storage Capacity for the Noise-Free Channel

An important property of DNA storage channels is the fact that the order or the molecules are lost. We first focus on this aspect of the channel model by studying the noise-free channel (where all copies are noise-free, i.e., the discrete memoryless channel is just the “identity channel”).

The main result of this section is the characterization of the storage capacity, given by the following theorem.

Theorem 1: The storage capacity of the noise-free shuffling-sampling channel is

$$C = (1 - q_0)(1 - 1/\beta). \tag{4}$$

In particular, if $\beta \leq 1$, no positive rate is achievable.

The capacity expression in (4) can be intuitively understood through the achievability argument. A storage rate of $R = (1 - q_0)(1 - 1/\beta)$ can be easily achieved by prefixing all the molecules with a distinct tag, which effectively converts the channel to a block-erasure channel. More precisely, we use the first $\log M$ bits of each molecule to encode a distinct index. Then we have $L - \log M = L(1 - 1/\beta)$ symbols left per molecule to encode data. The decoder can use the indices to remove duplicates and sort the molecules that are sampled. This effectively creates an erasure channel, where molecule i is erased if it is not drawn (i.e., $N_i = 0$) which occurs with probability q_0. Since the expected number of erasures is

$$\mathbb{E}\left[\frac{1}{M}\sum_{i=1}^{M} \mathbb{1}\{N_i = 0\}\right] = q_0,$$

we achieve storage rate

$$\frac{(1 - q_0)M(L - \log M)}{ML} = (1 - q_0)(1 - 1/\beta).$$

The surprising aspect of Theorem 1 is that this simple index-based scheme is optimal. It is also worth noting that the capacity expression only depends on the sampling distribution Q through the parameter q_0, i.e., the fraction of sequences that is *not seen* at the output of the channel.

In order to gain intuition on a practical implication of this theorem, suppose that each sequence is drawn according to a Poisson distribution with mean λ, so that in expectation λM sequences in total are drawn and λ can be thought of as the sequencing coverage depth. Then, the probability that a sequence is never drawn is $e^{-\lambda}$ and the capacity becomes

$$C = (1 - e^{-\lambda})(1 - 1/\beta). \tag{5}$$

This suggests that practical systems should not operate at a high coverage depth N/M, as high coverage depth significantly increases the time and cost of reading, but only provides little storage gains, according to our capacity expression. Notice that, in order to guarantee that all M sequences are observed at least once, we need $N = \Omega(M \log M)$ [9], [10]. When M is large, it is wasteful to operate in this regime, as this only gives a marginally larger storage capacity, but the sequencing costs can be exorbitant.

The result in Theorem 1 is flexible to allow different sampling models. In particular, one can consider separating the PCR amplification performed on each synthesized molecule from the sequencing step. Since one cannot control the PCR amplification factor precisely, it is reasonable to assume that a molecule x^L is first randomly amplified and a total of $A \geq 0$ copies are stored. If we consider a Poisson sampling model for the sequencing step, the effective coverage depth is $\lambda/\mathbb{E}[A]$ (since we are actually sampling from $M\mathbb{E}[A]$ molecules). In this case, the probability that none of the copies of x^L is sampled at the output is $\mathbb{E}\left[(e^{-\lambda/\mathbb{E}[A]})^A\right] = \mathbb{E}\left[(e^{(-\lambda/\mathbb{E}[A])A}\right]$. This can be recognized as the moment-generating function of A evaluated at $-\lambda/\mathbb{E}[A]$. In particular, when PCR is also modeled as a Poisson random variable with mean $\mathbb{E}[A] = \alpha$, $\mathbb{E}\left[e^{\theta A}\right] = e^{\alpha(e^{\theta}-1)}$, and the capacity of the resulting noise-free shuffling-sampling channel is

$$C = \left(1 - e^{-\alpha(1 - e^{-\lambda/\alpha})}\right)(1 - 1/\beta). \tag{6}$$

A. Motivation for Converse

A simple outer bound can be obtained by considering a genie that provides the decoder with the "true" index of each sampled molecule. In other words, $[x_1^L, \ldots, x_M^L]$ are the stored molecules, and the decoder observes $[y_1^L, \ldots, y_N^L]$ and the mapping $\sigma\colon \{1, \ldots, N\} \to \{1, \ldots, M\}$ so that $y_j^L = x_{\sigma(j)}^L$. This converts the channel into an erasure channel with block-erasure probability q_0, which yields

$$R \leq 1 - q_0. \tag{7}$$

It is intuitive that the bound (7) should not be achievable, as the decoder in general cannot sort the molecules and create an effective erasure channel. However, it is not clear a priori either whether prefixing every molecule with an index is optimal.

Notice that one can view the noise-free DNA storage channel as a channel where the encoder chooses a distribution (or a type) over the alphabet Σ^L and the decoder observes a noisy version of this type where the frequencies are perturbed accoding to Q. From this angle, the question becomes "how many types $t \in \mathbb{Z}_+^{2^L}$ with $\|t\|_1 = M$ can be reliably decoded?", and restricting ourselves to index-based schemes restricts the set of types to those with $\|t\|_\infty = 1$; i.e., no duplicate molecules are stored.

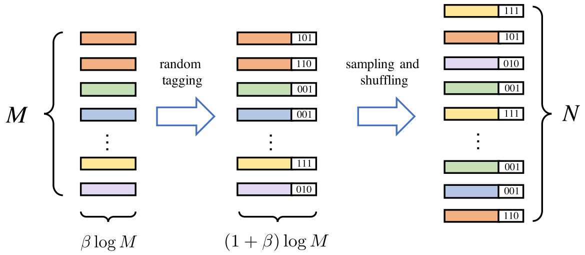

Fig. 2. Genie-aided channel for converse.

While this restriction may seem suboptimal, a counting argument suggests that it is not. The number of types for a sequence of length M over an alphabet of size $|\Sigma^L| = 2^L$ is at most M^{2^L} and thus at most

$$\frac{1}{ML} \log M^{2^L} = \frac{2^L \log M}{M\beta \log M} = \frac{M^\beta}{\beta M}$$

bits can be encoded per symbol. We conclude that, if $\beta < 1$, the capacity is $C = 0$. An actual bound on the rate can be obtained by counting the number of types more carefully. This is done in the following lemma, which we prove in the appendix.

Lemma 1: The number of distinct vectors $t \in \mathbb{Z}_+^a$ with $\|t\|_1 = b$ is given by

$$\mathcal{T}[a, b] := \binom{a+b-1}{b} < \left(\frac{e(a+b-1)}{b}\right)^b.$$

Since our types are vectors $t \in \mathbb{Z}_+^{2^L}$ with $\|t\|_1 = M$, and $2^L = 2^{\beta \log M} = M^\beta$, it follows that at most

$$\frac{1}{ML} \log \left(\frac{e(M^\beta + M - 1)}{M}\right)^M \leq \frac{M \log(\alpha M^{\beta-1})}{M\beta \log M}$$

bits can be encoded per symbol, for some $\alpha > 1$, and

$$R \leq 1 - 1/\beta. \tag{8}$$

Therefore, if we had a deterministic channel where the decoder observed *exactly* the M stored molecules, an index-based approach would be optimal from a rate standpoint. The converse presented in the next section utilizes a more careful genie to show that the bounds in (7) and (8) can in fact be combined, implying the optimality of index-based coding approaches.

B. Converse

Let $[x_1^L, \ldots, x_M^L]$ be the M length-L molecules written into the channel and $[y_1^L, \ldots, y_N^L]$ be the length-L molecules observed by the decoder. Notice that, whenever the channel output is such that $y_i^L = y_j^L$ for $i \neq j$, the decoder cannot determine whether both y_i^L and y_j^L were sampled from the same molecule x_ℓ^L or from two different molecules that obey $x_\ell^L = x_k^L, \ell \neq k$. In order to derive the converse, we consider a genie-aided channel that removes this ambiguity. As illustrated in Figure 2, before sampling the N molecules, the genie-aided channel appends a unique index of length $\log M$ to each molecule x_i^L, which results in the set of tagged molecules $\{(x_i^L, z_i)\}_{i=1}^M$. We emphasize that the indices z_i are all unique, and are chosen randomly and independently of the input sequences $\{x_i^L\}_{i=1}^M$. Notice that, in contrast to the naive genie discussed in Section III-A, this genie does *not* reveal the index i of the molecule x_i^L from which y_ℓ^L was sampled. Therefore, the channel is not reduced to an erasure channel, and intuitively the indices are only useful for the decoder to determine whether two equal samples $y_\ell^L = y_k^L$ came from the same molecule or from distinct molecules.

The output of the genie-aided channel, denoted by $\{(y_i^L, z_{\sigma(i)})\}_{i=1}^N$, is then obtained by sampling from the set of tagged molecules $\{(x_i^L, z_i)\}_{i=1}^M$, in the same way as the original channel samples the original molecules. The mapping $\sigma\colon [1:N] \to [1:M]$ is such that y_i^L was sampled from $x_{\sigma(i)}^L$. Notice that the actual mapping σ is not revealed to the decoder.

It is clear that any storage rate achievable in the original channel can be achieved on the genie-aided channel, as the decoder can simply discard the indices, or stated differently, the output of the original channel can be obtained from the output of the genie-aided channel.

Notice that $\{(y_i^L, z_{\sigma(i)})\}_{i=1}^N$ is in general a multiset. We let $\operatorname{set}(\{(y_i^L, z_{\sigma(i)})\}_{i=1}^N)$ be the set obtained from $\{(y_i^L, z_{\sigma(i)})\}_{i=1}^N$ by removing any duplicates. Then $\operatorname{set}(\{(y_i, z_{\sigma(i)})\}_{i=1}^N)$ is a sufficient statistic for $\{x_i^L\}_{i=1}^M$ since all tagged molecules are distinct objects, and sampling the same tagged molecule (x_i^L, z_i) does not yield additional information on $\{x_i^L\}_{i=1}^M$. More formally, conditioned on $\operatorname{set}(\{(y_i^L, z_{\sigma(i)})\}_{i=1}^N)$, $\{x_i^L\}_{i=1}^M$ is independent of the genie's channel output $\{(y_i^L, z_{\sigma(i)})\}_{i=1}^N$.

Next, we define the frequency vector $\mathbf{f} \in \mathbb{Z}_+^{M^\beta}$ (note that $|\Sigma^L| = 2^{\beta \log M} = M^\beta$) that is obtained from $\operatorname{set}(\{(y_i, z_{\tilde{i}})\}_{i=1}^N)$ in the following way. The entry of $\mathbf{f}$ corresponding to the molecule $y^L \in \Sigma^L$ is given by

$$\mathbf{f}[y^L] := \left|\{(y_j^L, z_{\sigma(j)}) \in \operatorname{set}(\{(y_i^L, z_{\sigma(i)})\}_{i=1}^N) \colon y_j^L = y^L\}\right|.$$

The frequency vector $\mathbf{f}$ is essentially a histogram that counts the number of occurrences of y^L in the set of tagged molecules $\{(y_i^L, z_{\sigma(i)})\}_{i=1}^N$. Notice that the entries of $\mathbf{f}$ can take values greater than one, because at the input we can choose to use the same molecule for multiple x_i^L.

Since $\operatorname{set}(\{(y_i^L, z_{\sigma(i)})\}_{i=1}^N)$ is a sufficient statistic for $\{x_i^L\}_{i=1}^M$ and the tags added by the genie were chosen at random and independently of $\{x_i^L\}_{i=1}^M$, it follows that $\mathbf{f}$ is also a sufficient statistic for $\{x_i^L\}_{i=1}^M$. Hence, we can view the (random) frequency vector $\mathbf{f}$ as the output of the channel without any loss. Notice that $|\operatorname{set}(\{(y_i^L, z_{\sigma(i)})\}_{i=1}^N)| = \|\mathbf{f}\|_1$, and we have $\|\mathbf{f}\|_1 \le M$ and $\mathbb{E}\left[\|\mathbf{f}\|_1/M\right] = 1 - q_0$. Furthermore, the following lemma asserts that $\|\mathbf{f}\|_1$ does not exceed its expectation by much.

Lemma 2: For any $\delta > 0$, the frequency vector $\mathbf{f}$ at the output of the genie-aided channel satisfies

$$\Pr\left(\frac{\|\mathbf{f}\|_1}{M} > 1 - q_0 + \delta\right) \to 0, \text{ as } M \to \infty.$$

Proof: Note that the number of distinct fragments that have been drawn is

$$\frac{\|\mathbf{f}\|_1}{M} = \frac{1}{M}\sum_{i=1}^M \mathbb{1}\{N_i > 0\}.$$

Since $\mathbb{1}\{N_i > 0\}$ are independent random variables with expectation $1 - q_0$, Hoeffding's inequality yields

$$\Pr\left(\frac{\|\mathbf{f}\|_1}{M} \ge (1 - q_0) + \delta\right) \le e^{-2M\delta^2},$$

which concludes the proof[1]. □

We now append the coordinate $f_0 = (1 - q_0 + \delta)M - \|\mathbf{f}\|_1$ to the beginning of $\mathbf{f}$ to construct $\mathbf{f}' = (f_0, \mathbf{f})$. Notice that when $\|\mathbf{f}\|_1 \le (1 - q_0 + \delta)M$ (which by Lemma 2 happens with high probability), we have $\|\mathbf{f}'\|_1 = (1 - q_0 + \delta)M$. This construction of $\mathbf{f}'$ will allow us to utilize Lemma 1 below. Fix $\delta > 0$, and define the event

$$\mathcal{E} = \{\|\mathbf{f}\|_1 > (1 - q_0 + \delta)M\} \tag{9}$$

with indicator function $\mathbb{1}_{\mathcal{E}}$. By Lemma 2, $\Pr(\mathcal{E}) \to 0$ as $M \to \infty$. Consider a sequence of codes $\{\mathcal{C}_M\}$ with rate R and vanishing error probability. If we let W be the message to be encoded, chosen uniformly at random from $\{1, \ldots, 2^{MLR}\}$. From Fano's inequality we have

$$\begin{aligned} MLR_s = H(W) &= I(W; \mathbf{f}') + H(W|\mathbf{f}') \\ &\le H(\mathbf{f}') + 1 + P_e MLR_s, \end{aligned} \tag{10}$$

where P_e is the probability of a decoding error, which by assumption goes to zero as $M \to \infty$. We can then upper bound the achievable storage rate R as

$$\begin{aligned} &MLR(1 - P_e) \le H(\mathbf{f}') + 1 \le H(\mathbf{f}', \mathbb{1}_{\mathcal{E}}) + 1 \\ &\le \Pr(\mathcal{E})\, H(\mathbf{f}' \,|\, \mathcal{E}) + \Pr(\bar{\mathcal{E}})\, H(\mathbf{f}' \,|\, \bar{\mathcal{E}}) + H(\mathbb{1}_{\mathcal{E}}) + 1. \end{aligned} \tag{11}$$

Note that the vector $\mathbf{f}'$ above has dimension $M^\beta + 1$ and, given the event $\bar{\mathcal{E}}$ occurs, $\|\mathbf{f}'\|_1 = (1 - q_0 + \delta)M$, and we have $H(\mathbf{f}'|\bar{\mathcal{E}}) \le \log \mathcal{T}[M^\beta + 1, (1 - q_0 + \delta)M]$, where $\mathcal{T}[a, b]$ is the number of vectors $x \in \mathbb{Z}_+^a$ with $\|x\|_1 = b$. From Lemma 1,

$$\begin{aligned} &\log \mathcal{T}[M^\beta + 1, (1 - q_0 + \delta)M] \\ &\quad \le (1 - q_0 + \delta)M \log\left(e + \frac{eM^{\beta-1}}{(1 - q_0 + \delta)}\right) \\ &\quad \le (1 - q_0 + \delta)M \log\left(\alpha M^{\beta-1}\right) \\ &\quad \le (1 - q_0 + \delta)M[(\beta - 1)\log M + \log \alpha], \end{aligned}$$

where α is a positive constant. Moreover, we notice that $\mathbf{f}'$ is a function of $\mathbf{f}$, which is a vector in $\mathbb{Z}_+^{M^\beta}$ with $\|\mathbf{f}\|_1 \le M$. Next, we define $\mathbf{f}'' = (f_0, \mathbf{f})$, again so that we can apply Lemma 1, where $f_0 = M - \|\mathbf{f}\|_1$ so that $\|\mathbf{f}''\|_1 = M$, and note that

$$\begin{aligned} H(\mathbf{f}'|\mathcal{E}) = H(\mathbf{f}''|\mathcal{E}) &\le \log \mathcal{T}[M^\beta + 1, M] \\ &\le M \log\left(\frac{e(M + M^\beta}{M}\right) \\ &\le M((\beta - 1)\log M + \log \alpha'), \end{aligned}$$

[1] An analogue of Lemma 2 can be proved for a different sampling model, which we describe in Appendix B.

where α' is another positive constant. Dividing (11) by ML and applying the bounds above yields

$$\begin{aligned}R(1-P_e) &\leq \Pr(\mathcal{E})\frac{M[(\beta-1)\log M+\log\alpha']}{ML}\\ &\quad+\frac{(1-q_0+\delta)M[(\beta-1)\log M+\log\alpha]}{ML}+\frac{2}{ML}\\ &\leq \Pr(\mathcal{E})\left(\frac{\beta-1}{\beta}+\frac{\log\alpha'}{\beta\log M}\right)\\ &\quad+(1-q_0+\delta)\left(1-\frac{1}{\beta}+\frac{\log\alpha}{\beta\log M}\right)+\frac{2}{ML}.\end{aligned}$$

Finally, letting $M\to\infty$ yields

$$R\leq(1-q_0+\delta)\left(1-1/\beta\right),$$

since $\Pr(\mathcal{E})\to 0$ by Lemma 2. Since $\delta>0$ can be chosen arbitrarily small, this concludes the converse proof.

IV. The Noisy Shuffling-Sampling Channel

Next, we study the effect of errors within sequences, in addition to the shuffling and sampling of the sequences. Instead of the general sampling distribution Q considered in Section III, we now focus on a simple choice of sampling distribution and let Q be distributed as Bernoulli$(1-q)$. Hence, a sequences is either drawn never or once, with the corresponding probabilities given by $\Pr(N_i=0)=q$ and $\Pr(N_i=1)=1-q$, for $i=1,\ldots,M$. Moreover, we assume that the molecules are all corrupted by a BSC with error probability p. We refer to this channel as the noisy shuffling-sampling channel.

A. The Capacity of the Noisy Shuffling-Sampling Channel

As in the error-free shuffling-sampling channel considered in Section III, we again consider a simple index-based coding scheme. As we will show, for a large set of parameters p and β, this scheme turns out to be capacity-optimal.

We consider a scheme based on an outer and an inner code and argue that it achieves a rate arbitrary close to

$$R_{\text{index}}=(1-q)(1-H(p)-1/\beta). \tag{12}$$

As outer code, we take a erasure-correcting code with block length M and rate $(1-q)$, where each symbol is itself a binary string of length $L(1-H(p)-1/\beta-\epsilon)$, for some small $\epsilon>0$. As inner code, we take a code designed for a BSC with codewords of length L and rate $R_{\text{BSC}}=1-H(p)-\epsilon$. We first encode the information using the outer code, which yields M symbols given as binary strings of length

$$L(1-H(p)-1/\beta-\epsilon)=LR_{\text{BSC}}-\log M.$$

We take each symbol, add a unique binary index of length $\log M$ and encode the resulting sequence using the BSC code, which yields M length-L sequences. With this scheme, we encode a total of $(1-q)M(LR_{\text{BSC}}-\log M)$ data bits, with a data rate of

$$\frac{(1-q)M\left(LR_{\text{BSC}}-\log M\right)}{ML}=(1-q)(R_{\text{BSC}}-1/\beta). \tag{13}$$

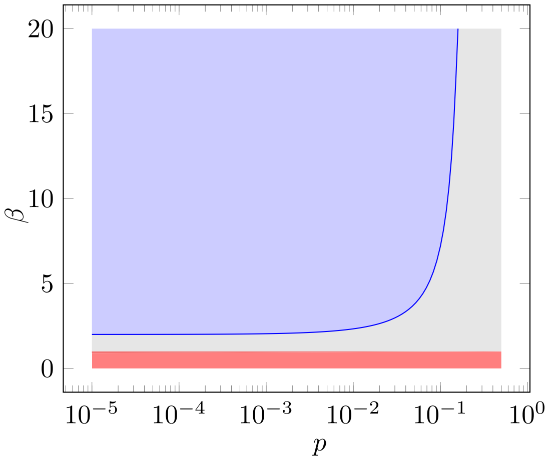

Fig. 3. Parameter regions for which the capacity is characterized. The capacity in the blue region is given by $C=(1-q)(1-H(p)-1/\beta)$, and the capacity in the red region (i.e., for $\beta<1$) is 0. In the gray region, it is still unknown.

Since $\epsilon>0$ can be chosen arbitrarily small, this scheme achieves a rate arbitrarily close to the rate given in (12), as claimed. For simplicity, in this short argument we did not take into account that the inner codeword is decoded with an error with a vanishing probability; we refer to Appendix C for a more formal achievability argument taking this into account.

On the other hand, the result from Section III, with $Q\sim\text{Ber}(1-q)$ implies that $C\leq(1-q)(1-1/\beta)$, since the error-free shuffling-sampling channel cannot be worse than the noisy shuffling-sampling channel. Furthermore, a simple genie-aided argument where the decoder observes the shuffling map can be used to establish that $C\leq(1-q)C_{\text{BSC}}$, where $C_{\text{BSC}}=1-H(p)$ is the capacity of a BSC with crossover probability p. Hence, a capacity upper bound is given by

$$C\leq(1-q)\min\left[1-H(p),1-1/\beta\right]. \tag{14}$$

Our main result improves on the upper bound in (14), and establishes that for parameters (p,β) in a certain regime, the lower bound in equation (12) is the capacity.

Theorem 2: For the noisy shuffling-sampling channel,

$$C=(1-q)(1-H(p)-1/\beta), \tag{15}$$

as long as $p<1/4$ and $1-H(2p)-2/\beta>0$. Moreover, if $\beta\leq 1$, the capacity is $C=0$.

The set of parameters (p,β) such that $1-H(2p)-2/\beta>0$ and $p<1/4$ is the blue region in Figure 3. In particular, (15) holds if $p\leq 0.1$ and $\beta\geq 6.4$, or if $p<0.01$ and $\beta\geq 2.35$.

B. Converse

To derive the converse, we view the input to the channel as a binary string of length ML, denoted by

$$X^{ML}=\left[X_1^L,X_2^L,\ldots,X_M^L\right]\in\{0,1\}^{ML}$$

or, equivalently, M strings of length L concatenated to form a single string of length ML. Similarly, the output of the channel is

$$Y^{NL}=\left[Y_1^L,Y_2^L,\ldots,Y_N^L\right]\in\{0,1\}^{NL},$$

where $N = \sum_i N_i$. It is useful to define a vector $S^N \in \{1, \ldots, M\}^N$ indicating the input string from which each output string was sampled. Furthermore, we let $Z^{NL} = \left[Z_1^L, \ldots, Z_N^L\right]$ be the random binary error pattern created by the BSC on the N non-deleted strings. We can now define the input-output relationship

$$Y_k^L = X_{S(k)}^L \oplus Z_k^L, \quad \text{for } k = 1, \ldots, N, \tag{16}$$

where $\oplus$ indicates elementwise modulo 2 addition. Note that the N_i's are fully determined by the vector S^N since $N_i = |\{i\colon S(k) = i\}|$. Also note that, since $Q \sim \text{Ber}(1-q)$, $N \leq M$ with probability 1.

Consider a sequence of codes for the noisy shuffling-sampling channel with rate R and vanishing error probability. Let $X^{ML} = \left[X_1^L, X_2^L, \ldots, X_M^L\right]$ be the input to the channel when we choose one of the 2^{MLR} codewords from one such code uniformly at random, and $Y^{NL} = \left[Y_1^L, Y_2^L, \ldots, Y_M^L\right]$ be the corresponding output. From Fano's inequality we have that $H(X^{ML}|Y^{ML}) \leq 1 + P_{e,M}ML \leq ML\epsilon_M$, where $P_{e,M}$ is the decoding error probability (of the code indexed by M) and $\{\epsilon_M\}$ is a sequence such that $\epsilon_M \to 0$ as $M \to \infty$. Thus,

$$MLR = H\left(X^{ML}\right) \leq I\left(X^{ML}; Y^{NL}\right) + ML\epsilon_M,$$

where $\epsilon_M \to 0$ as $M \to \infty$ by Fano's inequality. Then,

$$\begin{aligned}
ML(R - \epsilon_M) &= H\left(Y^{NL}\right) - H\left(Y^{NL}|X^{ML}\right) \\
&= H\left(Y^{NL}\right) - H\left(S^N, Z^{NL}, Y^{NL}|X^{ML}\right) \\
&\quad + H\left(S^N, Z^{NL}|X^{ML}, Y^{NL}\right) \\
&= H\left(Y^{NL}\right) - H\left(S^N, Z^{NL}, Y^{NL}|X^{ML}\right) \\
&\quad + H\left(S^N|X^{ML}, Y^{NL}\right)
\end{aligned} \tag{17}$$

The last equality follows by noticing that, given (S^N, X^{ML}, Y^{NL}), one can compute $Z_k^L = Y_k^L \oplus X_{S(k)}^L$ for $1 \leq k \leq N$, and thus $H\left(Z^{NL}|X^{ML}, Y^{NL}, S^N\right) = 0$. Since N is a function of S^N, and S^N and Z^{NL} are independent of X^{ML}, the second term in (17) can be expanded as

$$\begin{aligned}
&H\left(S^N, Z^{NL}, Y^{NL}|X^{ML}\right) \\
&= H\left(S^N|X^{ML}\right) + H\left(Z^{NL}|S^N, X^{ML}\right) \\
&\quad + H\left(Y^{NL}|X^{ML}, S^N, Z^{NL}\right) \\
&\overset{(i)}{=} H\left(S^N, N\right) + H\left(Z^{NL}|S^N, N\right) \\
&\quad + H\left(Y^{NL}|X^{ML}, S^N, Z^{NL}\right) \\
&\overset{(ii)}{=} H(N) + H\left(S^N|N\right) + H\left(Z^{NL}|N\right) \\
&\overset{(iii)}{=} H(N) + \sum_{n=1}^{M} \Pr(N = n)\left[\log \frac{M!}{(M-n)!} + nLH(p)\right] \\
&\overset{(iv)}{=} \sum_{n=1}^{M} \Pr(N = n)\left(n \log M + nLH(p)\right) + o(ML) \\
&= \mathbb{E}[N]M\left(\log M + LH(p)\right) + o(ML) \\
&= (1-q)\left[M \log M + MLH(p)\right] + o(ML).
\end{aligned} \tag{18}$$

In (i), we used the facts that S^N is independent of X^{ML}, N is a function of S^N, and Z^{NL} is independent of X^{ML} given S^N. Notice that X^{NL} is only dependent on S^N through N (which is a random variable). For (ii) we used that $H\left(Y^{NL}|X^{ML}, S^N, Z^{NL}\right) = 0$ since Y^{NL} is determined by X^{ML}, S^N, Z^{NL}, and (iii) follows from the fact that, given $N = n$, S^N is chosen uniformly at random from all vectors in $\{1, \ldots, M\}^n$ with distinct elements. For (iv), we used the fact that, from Stirling's approximation,

$$\begin{aligned}
\log \frac{M!}{(M-n)!} &= M \log M - (M-n)\log(M-n) + o(ML) \\
&= M \log M - (M-n)\log M \\
&\quad + (M-n)\log \frac{M}{M-n} + o(ML) \\
&= n \log M + (M-n)\log \frac{M}{M-n} + o(ML),
\end{aligned}$$

and, by Jensen's inequality,

$$\begin{aligned}
0 &\leq \sum_{n>0} \Pr(N = n)(M-n)\log \frac{M}{M-n} \\
&\leq (M - \mathbb{E}[N])\log \frac{M}{(M - \mathbb{E}[N])} \\
&= (1-q)M \log 1/q = o(ML).
\end{aligned}$$

In order to finish the converse, we need to jointly bound the first and third terms in equation (17). This step is summarized in the following lemma:

Lemma 3: If β and $p < 1/4$ satisfy

$$1 - H(2p) - 2/\beta > 0, \tag{19}$$

then it holds that

$$H\left(Y^{NL}\right) + H\left(S^N|X^{ML}, Y^{NL}\right) \leq (1-q)ML + o(ML).$$

The parameter regime (p, β) for which (19) holds is the regime in which our capacity expression holds, illustrated in Figure 3. Combining (17), (18) and Lemma 3, we have

$$\begin{aligned}
&ML(R - \epsilon_M) \\
&\quad \leq (1-q)\left(ML - MLH(p) - M \log M\right) + o(ML).
\end{aligned}$$

Dividing by ML and letting $M \to \infty$ yields the converse.

C. Intuition for Lemma 3

In order to discuss the intuition for Lemma 3 let us focus on the case $q = 0$; i.e., none of the molecules are lost at the output. In this case, $N = M$, and S^M is chosen uniformly at random from all permutations of $[1, ..., M]$. If we naively bound each entropy term separately, we obtain

$$H\left(Y^{ML}\right) + H\left(S^N|X^{ML}, Y^{ML}\right) \leq ML + M \log M.$$

However, intuitively, the bound $H\left(S^M|X^{ML}, Y^{ML}\right) \leq M \log M$ is too loose because, as we argue below, if the entropy term $H\left(Y^{ML}\right)$ is large then we expect $H\left(S^M|X^{ML}, Y^{NL}\right)$ to be small and vice versa.

To see this, first note that from $X^{ML} = x^{ML}$ and $Y^{ML} = y^{ML}$, one can estimate the permutation S that maps each output string to the corresponding input string, S^M, by finding, for each y_i^L, the x_j^L that is closest to it and setting $S(i) = j$. This is a good estimate if no other x_k^L is close to x_j^L. There are two regimes, illustrated in Figure 4, one

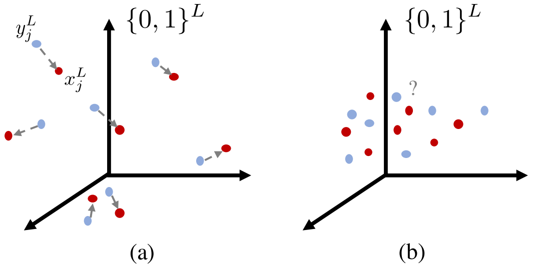

Fig. 4. Two opposite scenarios for estimating S^N from $\left(X^{ML}, Y^{NL}\right)$.

where S^N can be estimated well and one where it cannot. In the first regime, the strings $x_1^L, \ldots, x_M^L$ are all sufficiently distant from each other (in the Hamming sense). Hence, the maximum likelihood estimate of S^N given $X^{ML} = x^{ML}$ and $Y^{NL} = y^{ML}$ is "close" to the truth and we expect $H\left(S^N | X^{ML} = x^{ML}, Y^{NL} = y^{ML}\right)$ to be small. In the second regime, illustrated in Fig. 4(b), many of the sequences $x_1^L, \ldots, x_M^L$ are close to each other. So we have less information about S^N, and $H\left(S^N | X^{ML} = x^{ML}, Y^{NL} = y^{ML}\right)$ may be large.

On the other hand, the term $H\left(Y^{NL}\right)$ is maximized if the sequences $\left\{X_i^L\right\}$ are independent and if their values are uniformly distributed in $\{0,1\}^L$. Hence, in order for $H\left(Y^{NL}\right)$ to be large, we expect to be in the regime in Fig. 4(a) instead of the regime of Fig. 4(b). This leads to a tradeoff of the terms $H\left(Y^{NL}\right)$ and $H\left(S^N | X^{ML}, Y^{NL}\right)$, which we exploit to prove Lemma 3. The detailed proof, which considers the general case where $q \neq 0$, is presented in the appendix.

V. Discussion

In this paper we studied the fundamental limits of models of DNA-based storage systems, characterized by random sampling of the input sequences, shuffling, and perturbing them. Specifically, we considered a large class of channel models that capture a range of specific instances of DNA storage channels, specified by choices of synthesis, sequencing, and DNA handling technologies. We focused our analysis on two cases: (1) the error-free shuffling-sampling channel for an arbitrary sampling distribution Q and (2) the noisy shuffling-sampling channel where $Q \sim \text{Ber}(1-q)$ and the noisy channel is a BSC. In both cases we proved that a simple index-based scheme is capacity optimal, with the caveat that, for the noisy shuffling-sampling channel, the capacity expression in (15) only holds for the parameter regime of (p, β) in the blue region of Figure 3, and most importantly only holds in the low-error regime.

While the parameter regime in Figure 3 is arguably the most relevant one, an interesting question for future work is whether expression (15) is still the capacity of the BSC-shuffling channel if β and p do not satisfy (19) (i.e., the gray region in Figure 3). Notice that this is a high-noise, short-block regime, and it is reasonable to postulate that coding across the different sequences can be helpful and an index-based approach might not be optimal. Another natural question raised by Theorem 2 is whether a similar capacity expression holds for different noisy channels, including corruptions induced by deletions and insertions.

A. Beyond the Binary Symmetric Channel

Given that the capacity expression for the noisy shuffling-sampling channel given by (15) is $(1-q)$ $(C_{\text{BSC}} - 1/\beta)$, it is natural to ask whether for a different noisy channel with capacity C_{noisy}, the corresponding noisy shuffling-sampling channel has capacity $(1-q)(C_{\text{noisy}} - 1/\beta)$. Notice that, when the sampling distribution Q is $\text{Ber}(1-q)$, the index-based achievability scheme described in Section IV-A (and expanded on in Appendix C) can be extended in a straightforward way to achieve any rate below $(1-q)(C_{\text{noisy}} - 1/\beta)$. Hence, the challenging technical question is whether the converse argument can be generalized.

One class of channels for which the converse proof in Section IV-B can be extended are *symmetric* discrete memoryless channels (those channels are described in [20, Chapter 7.2]). For any channel in this class, the capacity-achieving input distribution is i.i.d. uniform of the input symbols and the resulting output distribution is i.i.d. uniform over the output symbols. Hence, the arguments in Section IV-B can be extended in a natural way. Specifically, for a noisy shuffling-sampling channel with sampling $Q \sim \text{Ber}(1-q)$, and a symmetric discrete memoryless channel (SDMC) with output alphabet $\mathcal{Y}$, we have:

Theorem 3: If β is large enough, the capacity of the SDMC shuffling-sampling channel is given by

$$C = (1-q)(C_{\text{SDMC}} - 1/\beta). \tag{20}$$

Moreover, if $\beta \leq \log|\mathcal{Y}|$, $C = 0$.

How large β needs to be for this statement to hold depends on the specific channel transition matrix. In order to go beyond symmetric channels, new converse techniques must be developed in order to establish a similar converse result.

To elaborate on this point, consider the simpler case where $N_i = 1$ with probability 1 for $i = 1, \ldots, M$, (i.e., all strings are sampled exactly once). Suppose we have an arbitrary channel $p(y^L | x^L)$ that maps length-L input strings to length-L output strings (which may not be memoryless) and capacity C_{noisy} (which requires the channel to be defined for $L \to \infty$). Consider the corresponding noisy shuffling channel. The index-based scheme achieves any rate $R < C_{\text{noisy}} - 1/\beta$. However, extending the proof in Section IV-B to establish $C_{\text{noisy}} - 1/\beta$ as the capacity is challenging. If we follow similar steps to those in (17), we have

$$\begin{aligned}
&ML(R - \epsilon_M) = I\left(X^{ML}; Y^{ML}\right) \\
&= H\left(Y^{NL}\right) - H\left(S^M, Y^{ML} | X^{ML}\right) \\
&\quad + H\left(S^M | X^{ML}, Y^{ML}\right) \\
&= H\left(Y^{NL}\right) - H\left(Y^{ML} | X^{ML}, S^M\right) - H\left(S^M\right) \\
&\quad + H\left(S^M | X^{ML}, Y^{ML}\right) \\
&= I\left(X^{ML}, S^M; Y^{ML}\right) - H\left(S^M\right) + H\left(S^M | X^{ML}, Y^{ML}\right).
\end{aligned}$$

Since $H(S^M) = M \log M + o(ML) = ML/\beta + o(ML)$, an outer bound to the noisy shuffling channel capacity in this general case is

$$C \leq \lim_{M \to \infty} \sup_{p(x^{ML})} \frac{I\left(X^{ML}, S^{M}; Y^{ML}\right)}{ML} + \frac{H\left(S^{M} | X^{ML}, Y^{ML}\right)}{ML} - 1/\beta. \quad (21)$$

The main challenge in establishing a general converse is the optimization over distributions of the channel input X^{ML}. If we consider distributions where $p(x^{ML}) = p(x^L) \times ... \times p(x^L)$ (i.e., independently encoding each of the input strings with the same $p(x^L)$), then the first term in the optimization becomes

$$\begin{aligned} \frac{I\left(X^{ML}, S^{M}; Y^{ML}\right)}{ML} &= \frac{H\left(Y^{ML}\right) - \left(Y^{ML} | X^{ML}, S^{M}\right)}{ML} \\ &= \frac{\sum_{i=1}^{M} H\left(Y_i^L\right) - \left(Y_i^L | X^L_{S(i)}\right)}{ML} \\ &= \frac{I(X^L; Y^L)}{L}, \end{aligned}$$

and by choosing the distribution $p(x^L)$ that achieves the capacity of the noisy channel $p(y^L|x^L)$, this term becomes C_{noisy}. In Section IV-A, we took advantage of the fact that, for a BSC, the input distribution is known, and the resulting output distribution $p(y^L)$ is i.i.d $\text{Ber}(1/2)$, in order to prove that this distribution maximizes the expression in (21). However, it is difficult to extend this for an arbitrary channel because of the second term $H\left(S^M | X^{ML}, Y^{ML}\right)$, particularly when the capacity-achieving distribution is unknown.

B. Independent Noisy Draws

Another interesting direction for extending the results in Section IV is to consider a noisy shuffling channel where the same input string can be observed multiple times at the output with independent noise patterns. One way to obtain such a channel is to consider the noisy shuffling-sampling channel model in Section II with $Q \sim \text{Poisson}(c)$ and a $\text{BSC}(p)$ as the discrete memoryless channel. This channel can be seen as a modification of the channel studied in Section IV where instead of observing either one or zero copies for each of the input strings, any positive number of copies can be observed.

For the special case of one input string ($M = 1$), this channel reduces to a *multi-draw* BSC, whose capacity was characterized by Mitzenmacher [21]. The input to a multi-draw BSC is a binary string x^n of length n and the output is the result of passing x^n D times through a $\text{BSC}(p)$ channel, where D can be modeled as a $\text{Poisson}(c)$ random variable (although other distributions can be used). The capacity of this channel was shown in [21] to be $\mathbb{E}\left[C_{D,p}\right]$, where $C_{d,p}$ is the capacity of the multi-draw BSC with a fixed number of observations d, which can be written in closed-form [18], [21].

Lenz *et al.* [18] recently showed that the capacity of the BSC shuffling channel with multi-draws is upper bounded as

$$C \leq \mathbb{E}\left[C_{D,p}\right] - \frac{1}{\beta}(1 - e^{-c}), \quad (22)$$

as long as $p \leq \frac{1}{8}$ and $1/\beta < 1 - H(4p)$, and subsequently showed that, if $\frac{1}{\beta} < \frac{1-H(4p)}{2}$, then (22) is indeed the capacity of the BSC shuffling channel [19] with multi-draws.

The achievability of this result is based on a random codebook construction. The decoder performs a greedy-like clustering of the output strings, and then uses typicality decoding based on a new notion of typicality between a set of d output strings and an input string. The proof of the upper bound relies on the noise level being relatively small, as it requires the output strings coming from the same input string to be close together and well separated from other strings so that those strings can be clustered with small error probability.

An important direction for future work is to study independent noisy draws in the regime where the noise is relatively large (i.e., p is large for the case of a BSC), as this is a relevant case in practice [7]. In this case, the output strings are not guaranteed to cluster at the output, and new techniques must be developed to establish the capacity.

C. Storage-Recovery Tradeoff

Most studies on DNA-based storage emphasize the storage rate (or storage density), while sequencing costs are disregarded. From a practical point of view, it is important to understand, for a given storage rate, how much sequencing is required for reliable decoding, as this determines the time and cost required for retrieving the data. Thus, characterizing the storage-recovery trade-off is of practical relevance.

One way to do this is to consider, in addition to the storage rate, the *recovery rate*, defined as the number of bits recovered per DNA base sequenced,

$$R_r := \frac{\log |\mathcal{C}|}{NL}. \quad (23)$$

In a practical setting, one can control the amount of sequencing performed, typically specified in terms of the coverage depth N/M. If we consider the error-free shuffling-sampling channel from Section III, in the case where Q is a Poisson distribution with mean λ, then $\lambda = N/M$ is the coverage depth, and one would like to choose a value of λ that achieves a good trade-off between storage rate and recovery rate.

If we let R_s be the storage rate (previously just R, see (3)), from Theorem 1 and the fact that $R_s = \lambda R_r$, the (R_s, R_r) feasibility region can be fully characterized.

Corollary 1: For the error-free shuffling-sampling channel with $Q \sim \text{Pois}(\lambda)$, rates (R_s, R_r) are achievable if and only if, for some $c > 0$,

$$\begin{aligned} R_s &\leq \left(1 - e^{-\lambda}\right)(1 - 1/\beta), \\ R_r &\leq \frac{1 - e^{-\lambda}}{\lambda}(1 - 1/\beta). \end{aligned}$$

This region is illustrated in Figure 5. This tradeoff suggests that a good operating point would be achieved by not trying to maximize the storage rate (which technically requires $\lambda \to \infty$). Instead, by using some modest coverage depth $\lambda = 1, 2, 3$, most of the storage rate $(63\%, 86\%, 95\%$, respectively) can be achieved. This is in contrast to what has been done in practical DNA storage systems that have been developed thus far, where the decoding phase utilizes very deep sequencing.

To be concrete, suppose we are interested in minimizing the cost of storing data on DNA. Synthesis costs are currently

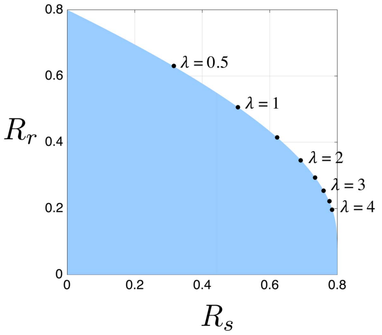

Fig. 5. (R_s, R_r) feasibility region for $\beta = 5$.

larger than sequencing costs by about a factor $q = 10^4$-10^5. Thus, if our goal is to minimize the cost for synthesizing and sequencing a given number of bits, the cost is proportional to $q/R_s + 1/R_r = \frac{q+\lambda}{1-e^{-\lambda}}$. This quantity can be minimized over λ, yielding the optimal cost per bit. For example, for $q = 10000$, $\lambda \approx 9.2$. Moreover, one may be interested in optimizing other quantities such as reading time or considering a scenario where the data is read more than once.

D. Storing Data on Short Molecules

Throughout this paper, we focused on the regime $L = \beta \log M$, with $\beta \geq 1$. For $\beta \leq 1$, no positive rate can be achieved (as shown by Theorem 1). However, motivated by the fact that it is in general much easier to synthesize very short sequences of DNA than longer ones, it is interesting to ask whether with very short sequences, it is still possible to build useful DNA storage systems.

Towards this goal, in this section we briefly discuss how fast the rate tends to zero in the regime when $\beta \leq 1$. Notice that, when $\beta \leq 1$, the total number of distinct molecules of length $L = \beta \log M$ is $2^{\beta \log M} = M^\beta < M$. Hence, it is impossible to write M distinct molecules. In this case, it is reasonable to study the amount of bits that can be stored relative to the number of potentially distinct molecules. Towards this goal we define the short-molecule rate $\tilde{R}$ as

$$\tilde{R} := \frac{\log |\mathcal{C}|}{M^\beta L}. \tag{24}$$

Proposition 1: Suppose that each molecule is drawn $N_i \sim Q$ times, with expectation $\mathbb{E}[N_i] > 0$, and that $\beta < 1$. Then, any achievable short-molecule rate satisfies $\tilde{R} \leq 1/\beta - 1$.

The proof, provided in the appendix, is based on the genie-aided and counting-based argument used in Section III-B. The proposition guarantees that the (true) rate R tends to zero at least as $1/M^{1-\beta}$. While at first sight, it might seem surprising that there is no dependency on $1 - q_0$, this is reasonable, since in the regime of $\beta < 1$, no more than M^β distinct molecules exist. Thus, we see each fragment about $\mathbb{E}[N]/M^\beta = \mathbb{E}[N_i] M/M^\beta = \mathbb{E}[N_i] M^{1-\beta}$ many times, which tends to infinity, regardless of Q.

We point out that index-based coding schemes cannot achieve the scaling $R = \Theta(M^\beta L)$ suggested by the proposition. To see this, suppose we encode the sequences by using $L - 1$ bits for the index and only one bit for the information, and repeat each such segment $M/(2^{L-1}) = 2M^{1-\beta}$ many times. We see each segment at least once with probability one as $M \to \infty$. Thus we reliably store $2^{L-1} = M^\beta/2$ bits. Simple variations of this scheme (where we change the number of bits allocated to the index) can be similarly shown to only encode $\Theta(M^\beta)$ bits reliably. Hence, for the regime $\beta \leq 1$, our upper bound to the number of bits that can be reliably stored is $\Theta(M^\beta L)$, while our lower bound is $\Theta(M^\beta)$, and it is an open question what the correct scaling is.

E. Outlook

In this paper we took steps towards the understanding of the fundamental limits of DNA-based storage systems. We proposed a simple model capturing the fact that molecules are stored in an unordered fashion, are short, and are corrupted by individual base errors. Our results show that a simple index-based coding scheme is asymptotically optimal for a large set of parameter choices.

While the model captures (moderate) substitution errors which are the prevalent error source on a nucleotide level of current DNA storage systems, the current generation of systems relies on *low-error* synthesis and sequencing technologies that are relatively expensive and limited in speed. A key idea towards developing the next-generation of DNA storage systems is to employ *high-error*, but cheaper and faster synthesis and sequencing technologies such as light-directed maskless synthesis of DNA and nanopore sequencing. Such systems induce a significant amount of insertion and deletion errors. Thus, and important area of further investigation is to understand the capacity of channels which introduce deletions and insertions as well.

Appendix A

Proof of Lemma 1: Notice that vectors $x \in \mathbb{Z}_+^a$ with $\|x\|_1 = b$ are in one-to-one correspondence with binary strings containing $(a-1)$ 0s and b 1s. For $x = (x_1, \ldots, x_a)$, the corresponding string is

$$\underbrace{1 \ldots 1}_{x_1} 0 \underbrace{1 \ldots 1}_{x_2} 0 \ldots 0 \underbrace{1 \ldots 1}_{x_a}. \tag{25}$$

It is clear that such a string has $(a-1)$ 0s and b 1s, and that distinct strings with $(a-1)$ 0s and b 1s correspond to distinct vectors x. The number of distinct strings of this form is

$$\frac{(a-1+b)!}{(a-1)!\, b!} = \binom{a+b-1}{b}.$$

The upper bound in the statement of the lemma is a standard bound for binomial coefficients. □

Appendix B
Proof of Lemma 2 Under a Sampling-With-Replacement Model

As it turns out, Lemma 2 can be proved under a sampling-with-replacement model. Under this model, instead of sampling each molecule according to a probability distribution Q, N sequences are sampled out of the pool of M stored sequences. Since there are multiple copies of each molecule in the pool due to PCR, we consider a sampling with replacement model. By proving Lemma 2 in this setting, one can establish a version of Theorem 1 for the sampling-with-replacement shuffling channel, as previously described in [22].

Consider the same genie-based argument described in Section III-B. In the sampling-with-replacement setting, the ℓ_1 norm of the frequency vector $\mathbf{f}$ at the output of the genie-aided channel is distributed as the number of distinct coupons obtained by drawing $N = \lambda M$ times with replacement from a set of M distinct coupons. Thus, Lemma 2 is an immediate consequence of the following stronger statement.

Lemma 4: Let Q be the number of distinct coupons obtained by drawing $N = \lambda M$ times with replacement from a set of M distinct coupons. We have that, for any $\delta > 0$,

$$\Pr\left(Q \geq (1 - e^{-\lambda} + \delta)M\right) \leq \frac{1}{M} \frac{2e^{2\lambda}}{2\left(\ln\left(\frac{e^{-\lambda}}{e^{-\lambda}-\delta}\right) - \frac{e^{\lambda}}{M}\right)^2}.$$

Proof: Since $\Pr\left(Q \geq (1 - e^{-\lambda} + \delta)M\right)$ is a non-increasing function of δ, we can assume that $\delta \in (0, e^{-\lambda}/2]$, as that simplifies the expressions. Let t_i be the number of draws to collect the i-th coupon after $(i-1)$ coupons have been collected, $i = 0, \ldots, M-1$, and consider the number of draws for obtaining αM distinct coupons $T := \sum_{i=0}^{\alpha M-1} t_i$ where $\alpha := 1 - e^{-\lambda} + \delta$. Due to

$$\Pr\left(Q \geq (1 - e^{-\lambda} + \delta)M\right) = \Pr\left(Q \geq \alpha M\right) = \Pr\left(T \leq N\right),$$

the lemma will follow by upper-bounding $\Pr\left(T \leq N\right)$ using Chebyshev's inequality. We first note that with $\mathbb{E}\left[t_i\right] = 1/p_i, p_i := \frac{M-i}{M}$ and $\operatorname{Var}\left[t_i\right] = \frac{1-p_i}{p_i^2}$, we obtain

$$\begin{aligned}\mathbb{E}\left[T\right] &= \sum_{i=0}^{\alpha M-1} \mathbb{E}\left[t_i\right] = M \sum_{i=0}^{\alpha M-1} \frac{1}{M-i}\\ &= M(H_M - H_{M(1-\alpha)})\\ &\geq M(\ln M - \ln(M(1-\alpha))) - \frac{1}{2(1-\alpha)}\\ &\geq -M\ln(1-\alpha) - e^{\lambda} = -M\ln(e^{-\lambda}-\delta) - e^{\lambda}\\ &= M\lambda + M\underbrace{\ln\left(\frac{e^{-\lambda}}{e^{-\lambda}-\delta}\right)}_{\xi} - e^{\lambda} = N + M\xi - e^{\lambda}.\end{aligned}$$

Here, $H_M = \sum_{i=1}^{M} \frac{1}{i}$ is the M-th harmonic number, and the first inequality follows by the asymptotic expansion

$$0 \leq H_n - \ln n - \gamma = \frac{1}{2n} - \frac{1}{12\, n^2} + \frac{1}{120\, n^4} - \ldots \leq \frac{1}{2n},$$

where γ is the Euler-Mascheroni constant. The second inequality follows from $\frac{1}{1-\alpha} \leq \frac{1}{e^{-\lambda}-e^{-\lambda}/2} = 2\, e^{\lambda}$. Moreover, the variance can be upper-bounded as

$$\begin{aligned}\operatorname{Var}\left[T\right] &= \sum_{i=0}^{\alpha M-1} \operatorname{Var}\left[t_i\right] = \sum_{i=0}^{\alpha M-1} \frac{iM}{(M-i)^2}\\ &\leq M\frac{\alpha}{2(1-\alpha)^2} \leq M2e^{2\lambda}.\end{aligned} \tag{26}$$

Using the bound on the expectation and Chebyshev's inequality, we have for any $\beta > 0$, that

$$\begin{aligned}&\Pr\left(-T + N + M\xi - e^{\lambda} > \beta\right)\\ &\qquad \leq \Pr\left(-T + \mathbb{E}\left[T\right] > \beta\right) \leq \frac{\operatorname{Var}\left[T\right]}{\beta^2}.\end{aligned}$$

Choosing $\beta = M\xi - e^{\lambda}$ and using the upper bound on $\operatorname{Var}\left[T\right]$ given in (26), yields $\Pr\left(T \leq N\right) \leq \frac{1}{M}\frac{2e^{2\lambda}}{\left(\xi - \frac{e^{\lambda}}{M}\right)^2}$, which concludes the proof. □

Appendix C
Achievability of Theorem 2

In this section, we give a formal argument for achievability of Theorem 2. Strictly speaking, the simple index-based scheme described in Section IV-A, based on a BSC inner code and an erasure outer code must be modified in order to formally prove the achievability of Theorem 2. In particular, we need to account for the fact that, if an inner codeword is decoded in error–which occurs with a vanishing probability–its unique index will also be decoded in error, likely causing an "index collision" with another correctly decoded inner codeword. Moreover, it is possible (although unlikely) that inner codewords are decoded in error but the decoded codewords have valid indices in a way that does not cause an erasure to occur, but rather a substitution error. Hence, instead of using an "off-the-shelf" outer code for an erasure channel, we consider a random code construction for outer code. Next we describe the argument in detail.

Our inner code is a code designed for a BSC with codewords of length L and rate $R_{\text{BSC}} = 1 - H(p) - \epsilon_1$ for some $\epsilon_1 > 0$. Our outer code is a code with $2^{ML(1-q-\epsilon_2)(R_{\text{BSC}}-1/\beta)}$ codewords of length M over the alphabet $\mathcal{A} = \{1, ..., 2^{LR_{\text{BSC}}-\log M}\}$, for some $\epsilon_2 > 0$. We consider a random codebook construction for the outer code, where each codeword has its symbols drawn independently and uniformly at random from $\mathcal{A}$.

We encode the $ML(1-q-\epsilon_2)(R_{\text{BSC}}-1/\beta)$ information bits first using the outer code, which yields M symbols from $\mathcal{A}$, each of which can be seen as a binary string of length $LR_{\text{BSC}} - \log M$. To each of these symbols, we append a unique binary index of length $\log M$, yielding M binary strings of length LR_{BSC}. Each of these can then be encoded using the BSC inner code into a binary string of length L. The resulting M strings are the input to the noisy shuffling channel.

The decoder operates as follows. For each output string of length L we apply the decoder from the BSC code. If the decoder does not return a codeword, we discard that string. If it returns a codeword, we recover its LR_{BSC} information bits. The first $\log M$ bits are treated as an index, and the remaining $LR_{\text{BSC}} - \log M$ bits are converted to a symbol in $\mathcal{A}$. The indices can then be used to sort the recovered symbols from $\mathcal{A}$. If two decoded symbols have the same index (which must have been due to a decoding error in one or both of the inner symbols) we discard both and treat them as erasures. The symbol associated with a missing index is also declared as an erasure. As a result, we obtain an output sequence of length M over the alphabet $\mathcal{A} \cup \{\varepsilon\}$, where ε is the erasure symbol. Let $Y_{\mathcal{A}} \in (\mathcal{A} \cup \{\varepsilon\})^M$ be this output sequence. If there exists a unique codeword from the outer codebook that matches $Y_{\mathcal{A}}$ in at least $M(1-q-\epsilon_2/2)$ positions, we return it. Otherwise, the decoder declares an error.

Next we analyze the error probability of this code, averaged over all possible outer codes and conditioned on the fact that codeword 1 is chosen (without loss of generality). We need to consider two types of error events: the event $\mathcal{E}_1$ that $Y_{\mathcal{A}}$ does not match codeword 1 in at least $M(1-q-\epsilon_2/2)$ positions, and the event $\mathcal{E}_2$ that a codeword other than 1 matches $Y_{\mathcal{A}}$ in at least $M(1-q-\epsilon_2/2)$ positions. Let $Z \in \{1, ..., M\}$ be the total number of output sequences (out of the N observed at the output) that are decoded in error. To bound $\Pr(\mathcal{E}_1)$ we notice that, if at least $M(1-q-\epsilon_2/4)$ length-L strings are observed at the output (i.e., $N \geq M(1-q-\epsilon_2/4)$) and $Z \leq M\epsilon_2/8$, then at least $M(1-q-\epsilon_2/4) - 2(M\epsilon_2/8) = M(1-q-\epsilon_2/2)$ symbols of $Y_{\mathcal{A}}$ are correct symbols from codeword 1 (the factor of 2 before $(M\epsilon_2/8)$ is to account for possible index collisions). Hence, if this occurs, the output sequence (after inner decoding) will match the true codeword in at least $M(1-q-\epsilon_2/2)$ positions. We can thus bound $\Pr(\mathcal{E}_1)$ as

$$\begin{aligned}\Pr(\mathcal{E}_1) &\leq \Pr\left(N < M(1-q-\epsilon_2/4)\right)\\ &\quad + \Pr\left(Z > M\epsilon_2/8 \,|\, N \geq M(1-q-\epsilon_2/4)\right).\end{aligned}$$

From the symmetry and independence of all output strings, Hoeffding's inequality implies that

$$\begin{aligned}&\Pr\left(Z > M\epsilon_2/8 \,|\, N \geq M(1-q-\epsilon_2/4)\right)\\ &\leq \exp\left[-2M(1-q-\epsilon_2/4)\left(\frac{\epsilon_2/8}{1-q-\epsilon_2/4} - P_{\text{BSC},e,M}\right)^2\right]\end{aligned}$$

where $P_{\text{BSC},e,M}$ is the error probability of the inner BSC code (for blocklength $L = \beta \log M$). Since $P_{\text{BSC},e,M} \to 0$ as $M \to \infty$, we see that the above bound tends to 0 as $M \to \infty$, for ϵ_1 and ϵ_2 small enough. A straightforward application of Hoeffding's inequality to bound $\Pr\left(N < M(1-q-\epsilon_2/4)\right)$ then implies that $\Pr(\mathcal{E}_1) \to 0$ as $M \to \infty$.

In order to bound $\Pr(\mathcal{E}_2)$, we need to bound the probability that another codeword matches $Y_{\mathcal{A}}$ in at least $M(1-q-\epsilon_2/2)$ positions. Notice that an incorrect inner decoding may lead to an erasure or an incorrect symbol (although the latter occurs with smaller probability). If $Z \leq M\epsilon_2/8$, then $\mathcal{E}_2$ can only happen if another codeword shares at least

$$M(1-q-\epsilon_2/2) - M\epsilon_2/8 = M(1-q-5\epsilon_2/8)$$

symbols of codeword 1. From a union bound over all other $2^{ML(1-q-\epsilon_2)(R_{\text{BSC}}-1/\beta)} - 1$ codewords, this happens with probability at most

$$\begin{aligned}&2^{ML(1-q-\epsilon_2)(R_{\text{BSC}}-1/\beta)}(1/|\mathcal{A}|)^{M(1-q-5\epsilon_2/8)}\\ &= 2^{ML(1-q-\epsilon_2)(R_{\text{BSC}}-1/\beta)}2^{-(LR_{\text{BSC}}-\log M)M(1-q-5\epsilon_2/8)}\\ &= 2^{ML(1-q-\epsilon_2-(1-q-5\epsilon_2/8))(R_{\text{BSC}}-1/\beta)}\\ &= 2^{-ML(R_{\text{BSC}}-1/\beta)(3\epsilon_2/8)},\end{aligned}$$

which tends to 0 as $M \to \infty$. The arguments used to bound $\Pr(\mathcal{E}_1)$ can again be used to bound $\Pr(Z > M\epsilon_2/8)$, and we conclude that $\Pr(\mathcal{E}_2) \to 0$ as $M \to \infty$. We conclude that rate

$$\begin{aligned}&(1-q-\epsilon_2)(R_{\text{BSC}} - 1/\beta)\\ &\quad = (1-q-\epsilon_2)(1-H(p)-\epsilon_1-1/\beta)\end{aligned}$$

is achievable for any $\epsilon_1, \epsilon_2 > 0$, concluding the formal achievability proof of Theorem 2.

Appendix D
Proof of Lemma 3

Let $Y_1^L, \ldots, Y_N^L$ be the N strings observed at the output of the channel. First we notice that, since N is a function of Y^{NL}, we can write

$$\begin{aligned}&H\left(Y^{NL}\right) + H\left(S^N|X^{ML}, Y^{NL}\right)\\ &= H\left(Y^{NL}, N\right) + H\left(S^N|X^{ML}, Y^{NL}, N\right)\\ &= H(N) + H\left(Y^{NL}|N\right) + H\left(S^N|X^{ML}, Y^{NL}, N\right)\\ &= H(N) + \sum_{n>0}\Pr(N=n)\left[H\left(Y^{NL}|N=n\right)\right.\\ &\qquad\left. + H\left(S^N|X^{ML}, Y^{NL}, N=n\right)\right]. \qquad (27)\end{aligned}$$

We will show that

$$\begin{aligned}&H\left(Y^{NL}|N=n\right) + H\left(S^N|X^{ML}, Y^{NL}, N=n\right)\\ &\leq nL + n\log\frac{M}{n} + o(ML), \qquad (28)\end{aligned}$$

which, when plugged back into (27) implies that

$$\begin{aligned}&H\left(Y^{NL}\right) + H\left(S^N|X^{ML}, Y^{NL}\right)\\ &\qquad \leq \mathbb{E}[N]L + \mathbb{E}[N\log M/N] + o(ML)\\ &\qquad \leq (1-q)ML + o(ML), \qquad (29)\end{aligned}$$

where we used the fact that $H(N) = o(ML)$, $\mathbb{E}[N] = (1-q)M$, and Jensen's inequality applied to the concave function $x\log(M/x)$. This will establish the lemma.

In order to capture whether we are in the regime of Figure 4(a) or (b), we let T be the largest subset of $[1:n]$ so that, for any $i,j \in T$, $d_H\left(Y_i^L, Y_j^L\right) \geq \alpha L$, where d_H is the Hamming distance and $\alpha > 2p$. We assume that in case of ties, an arbitrary tie-breaking rule is used to define T (the actual choice will not be relevant for the proof).

Let $\mathbb{E}_n$ be the expectation conditioned on $N = n$; i.e., $\mathbb{E}_n[\cdot] = \mathbb{E}[\cdot|N=n]$. We prove that, given the conditions in Lemma 3, the following two bounds involving $\mathbb{E}_n|T|$ hold:

$$\begin{aligned}&\text{(B1)}\ H\left(Y^{NL}|N=n\right) \leq L\mathbb{E}_n|T|\\ &\qquad + (n-\mathbb{E}_n|T|)\left(\log\mathbb{E}_n|T| + LH(\alpha)\right) + o(ML), \quad \text{(30a)}\\ &\text{(B2)}\ H\left(S^N|X^{ML}, Y^{NL}, N=n\right) \leq n\log M\\ &\qquad - \mathbb{E}_n|T|\log\mathbb{E}_n|T| + o(ML). \qquad \text{(30b)}\end{aligned}$$

For large $\mathbb{E}_n|T|$, we are typically in the regime of Figure 4(a), while Figure 4(b) corresponds to the case where $\mathbb{E}_n|T|$ is small. The bounds above capture the tension between the terms $H\left(Y^{NL}|N=n\right)$ and $H\left(S^N|X^{ML}, Y^{NL}, N=n\right)$ because (B2) is decreasing in $\mathbb{E}_n|T|$, while (B1) is increasing in $\mathbb{E}_n|T|$ (provided that $\beta(1-H(\alpha)) \geq 1$). Combining (B1) and (B2),

$$\begin{aligned} &H\left(Y^{NL}|N=n\right) + H\left(S^N|X^{ML}, Y^{NL}, N=n\right) \\ &\quad \leq L\mathbb{E}_n|T| + (n-\mathbb{E}_n|T|)\left(\log \mathbb{E}_n|T| + LH(\alpha)\right) \\ &\qquad + n\log M - \mathbb{E}_n|T|\log \mathbb{E}_n|T| + o(ML) \\ &\quad = \mathbb{E}_n|T|L(1-H(\alpha)) + n\log \mathbb{E}_n|T| - 2\mathbb{E}_n|T|\log \mathbb{E}_n|T| \\ &\qquad + nLH(\alpha) + n\log M + o(ML). \end{aligned} \tag{31}$$

Replacing $\mathbb{E}_n|T|$ with x and ignoring the terms in this upper bound that do not involve x, we have the expression

$$f(x) \triangleq \gamma x \log M + n\log x - 2\ x\log x,$$

where we define $\gamma = \beta(1-H(\alpha))$. For $x > 0$, we have

$$\begin{aligned} f'(x) &= \frac{1}{\ln(2)}\left(\gamma \ln M + \frac{n}{x} - 2\ln x - 2\right) \\ &> \frac{1}{\ln(2)}\left(\gamma \ln M - 2\ln x - 2\right) \\ &= \frac{2}{\ln(2)}\left(\ln \frac{M^{\gamma/2}}{x} - 1\right). \end{aligned}$$

Hence $f'(x) > 0$ if

$$x < e^{-1}M^{\gamma/2}. \tag{32}$$

We see that, as long as $\gamma > 2$, the right-hand side of (32) is greater than M for M large enough. This means that $f(x)$ is increasing for $1 \leq x \leq M$. Since $\mathbb{E}_n|T| \leq n \leq M$, f must attain its maximum at $f(n)$. Therefore, (31) can be upper-bounded by setting $x = \mathbb{E}_n|T| = n$, which yields

$$\begin{aligned} &H\left(Y^{NL}|N=n\right) + H\left(S^N|X^{ML}, Y^{NL}, N=n\right) \\ &\qquad \leq nL + n\log\frac{M}{n} + o(ML). \end{aligned}$$

Notice that this holds if, for some $\alpha > 2p$,

$$\gamma = \beta(1-H(\alpha)) > 2 \Leftrightarrow 1 - H(\alpha) - 2/\beta > 0.$$

From the continuity of $H(\cdot)$, such α can be found if (19) holds, proving the lemma. It remains to prove (B1) and (B2).

Proof of (B1): Since T is a deterministic function of Y^{NL} and can take at most 2^n values,

$$\begin{aligned} &H\left(Y^{NL}|N=n\right) = H\left(Y^{NL}, T|N=n\right) \\ &\quad = H\left(T|N=n\right) + H\left(Y^{NL}|T, N=n\right) \\ &\quad \leq n + \sum_{t\subseteq[1:n]} \Pr\left(T=t|N=n\right) H\left(Y^{NL}|T=t, N=n\right). \end{aligned} \tag{33}$$

Next we notice that, for a given t, we can write

$$\begin{aligned} &H\left(Y^{NL}|T=t, N=n\right) = H\left([Y_i^L : i\in t]|T=t, N=n\right) \\ &\quad + H\left([Y_i^L : i\notin t]|T=t, N=n, [Y_i^L : i\in t]\right). \end{aligned} \tag{34}$$

The first term in (34) is trivially bounded as

$$H\left([Y_i^L : i\in t]|T=t, N=n\right) \leq |t|L.$$

Each of the remaining length-L strings Y_i^L with $i\notin t$ must be within a distance αL from one of the strings in $[Y_i^L : i\in t]$, from the definition of T. Hence, conditioned on $[Y_i^L : i\in t]$, each of them can only take at most $|t||B(\alpha L)|$ values, where $B(\alpha L)$ is a Hamming ball of radius αL. Since $|B(\alpha L)| \leq 2^{LH(\alpha)}$ for $\alpha < 1/2$, we bound the second term in (34) as

$$\begin{aligned} &H\left([Y_i^L : i\notin t]|T=t, N=n, [Y_i^L : i\in t]\right) \\ &\qquad \leq (n-|t|)\left(\log|t| + LH(\alpha)\right). \end{aligned} \tag{35}$$

We point out that, for large α, we may have $|t||B(\alpha L)| > 2^L$, making (35) a loose bound, but good enough for our purposes. Using these bounds back in (33), we obtain

$$\begin{aligned} &H\left(Y^{NL}|N=n\right) \\ &\quad \leq n + \mathbb{E}_n\left[L|T| + (n-|T|)\left(\log|T| + LH(\alpha)\right)|N=n\right] \\ &\qquad + o(ML) \\ &\quad \leq L\mathbb{E}_n|T| + (n-\mathbb{E}_n|T|)\left(\log \mathbb{E}_n|T| + LH(\alpha)\right) + o(ML), \end{aligned} \tag{36}$$

where we used the fact that $(n-x)\log x$ is a concave function of x and Jensen's inequality. □

Proof of (B2): Since T is a deterministic function of Y^{NL},

$$\begin{aligned} &H\left(S^N|X^{ML}, Y^{NL}, N=n\right) \\ &\quad = H\left(S^N|X^{ML}, Y^{NL}, T, N=n\right) \\ &\quad = \sum_{t\subseteq[1:n]} \Pr\left(T=t|N=n\right) \\ &\qquad\qquad \times H\left(S^N|X^{ML}, Y^{NL}, T=t, N=n\right) \\ &\quad \leq \sum_{t\subseteq[1:n]} \Pr\left(T=t|N=n\right) \\ &\qquad\qquad \times \sum_{i=1}^{n} H\left(S(i)|X^{ML}, Y^{NL}, T=t, N=n\right). \end{aligned} \tag{37}$$

Next we notice that the probability that δL or more errors occur in a single length-L string, for $\delta > p$, is at most $2^{-LD(\delta\|p)}$ by the Chernoff bound (where $D(\cdot\|\cdot)$ is the binary KL divergence). If we let $\mathcal{E}_i$ be the event that $d_H\left(X_{S(i)}^L, Y_i^L\right) \geq \delta L$, then we have

$$\Pr(\mathcal{E}_i) \leq 2^{-LD(\delta\|p)} = M^{-\beta D(\delta\|p)}.$$

The conditional entropy term in (37) is upper bounded as

$$\begin{aligned} &H\left(S(i)|X^{ML}, Y^{NL}, T=t, N=n\right) \\ &\quad \leq H\left(S(i), \mathbb{1}_{\mathcal{E}_i}|X^{ML}, Y^{NL}, T=t, N=n\right) \\ &\quad \leq H(\mathbb{1}_{\mathcal{E}_i}|T=t, N=n) + \Pr(\mathcal{E}_i|T=t, N=n) \\ &\qquad\quad \times H\left(S(i)|X^{ML}, Y^{NL}, T=t, N=n, \mathcal{E}_i\right) \\ &\qquad + \Pr(\bar{\mathcal{E}}_i|T=t, N=n) \\ &\qquad\quad \times H\left(S(i)|X^{ML}, Y^{NL}, T=t, N=n, \bar{\mathcal{E}}_i\right) \\ &\quad \leq 1 + \Pr(\mathcal{E}_i|T=t, N=n)\log M \\ &\qquad + H\left(S(i)|X^{ML}, Y^{NL}, T=t, N=n, \bar{\mathcal{E}}_i\right). \end{aligned} \tag{38}$$

The final step is to bound the conditional entropy term in (38), for the case where $i\in t$. Set $\delta = \alpha/2$. Conditioned on $\bar{\mathcal{E}}_i$, $d_H\left(X_{S(i)}^L, Y_i^L\right) < \alpha L/2$. Moreover, conditioned on $T=t$, for any $j\in t-\{i\}$, $d_H\left(Y_i^L, Y_j^L\right) \geq \alpha L$. For $i\in t$, we define

$$A_i = \{j : Y_i^L \text{ is the closest output string in } t \text{ to } X_j^L\}.$$

Notice that A_i, $i \in t$, forms a partition of $[1:M]$. We claim that, if $i \in t$, $S(i)$ must be in A_i. To see this notice that, for any $k \in t$, $k \neq i$, we have

$$\begin{aligned}\alpha L &\leq d_H\left(Y_i^L, Y_k^L\right)\\ &\leq d_H\left(X_{S(i)}^L, Y_i^L\right) + d_H\left(X_{S(i)}^L, Y_k^L\right)\\ &< \alpha L/2 + d_H\left(X_{S(i)}^L, Y_k^L\right),\end{aligned}$$

implying that $d_H\left(X_{S(i)}^L, Y_k^L\right) > \alpha L/2 \geq d_H\left(X_{S(i)}^L, Y_i^L\right)$, and thus $S(i) \in A_i$. Therefore, $S(i)$ for each output string Y_i^L with $i \in t$, can take at most $|A_i|$ values. Hence we have

$$\begin{aligned}&\sum_{i=1}^{n} H\left(S(i)|X^{ML}, Y^{NL}, T=t, N=n, \bar{\mathcal{E}}_i\right)\\ &\quad\leq \sum_{i\notin t}\log M + \sum_{i\in t}\log|A_i|\\ &\quad= (n-|t|)\log M + \sum_{i\in t}\log|A_i|\\ &\quad\leq (n-|t|)\log M + |t|\log(M/|t|)\\ &\quad= n\log M - |t|\log|t|, \qquad (39)\end{aligned}$$

where the last inequality follows because $\sum_{i\in t}|A_i| = M$, and the sum is maximized by $|A_i| = M/|t|$. Combining (37), (38), and (39), we obtain

$$\begin{aligned}&H\left(S^N|X^{ML}, Y^{NL}, N=n\right)\\ &\leq \sum_{i=1}^{n}\sum_{t\subseteq[1:n]}\Pr\left(T=t|N=n\right)\\ &\qquad\times\left[1+\Pr(\mathcal{E}_i|T=t,N=n)\log M\right]\\ &\quad+\sum_{t\subseteq[1:n]}\Pr\left(T=t|N=n\right)\\ &\qquad\times\sum_{i=1}^{n}H\left(S(i)|X^{ML},Y^{NL},T=t,N=n,\bar{\mathcal{E}}_i\right)\\ &= n+\log M\sum_{i=1}^{n}\Pr(\mathcal{E}_i|N=n)+n\log M-\mathbb{E}_n\left[|T|\log|T|\right]\\ &\overset{(i)}{\leq} n+M\log M\Pr(\mathcal{E}_i)+n\log M-\mathbb{E}_n\left[|T|\log|T|\right]\\ &\overset{(ii)}{\leq} n+M^{-\beta D(\delta\|p)}M\log M+n\log M-\mathbb{E}_n|T|\log\mathbb{E}_n|T|\end{aligned}$$

where, in (i) we used the fact that $\mathcal{E}_i$ is independent of $N=n$ and $n \leq M$, and in (ii) we used Jensen's inequality. Since $M^{-\beta D(\delta\|p)} \to 0$ as $M \to \infty$, $M^{-\beta D(\delta\|p)}M\log M = o(ML)$, concluding the proof. □

Appendix E

Proof of Proposition 1: We use a similar genie-aided and counting-based proof as in Section III-B. The only difference is on how the number of frequency vectors is bounded. As before, the frequency vector on the output of the genie-aided channel satisfies, for any $\delta > 0$, $\|\mathbf{f}\|_1 \leq M(1-q_0+\delta)$. We next upper bound the number of different frequency vectors $\mathbf{f} \in \mathbb{Z}_+^{M^\beta}$ with $\|\mathbf{f}\|_1 = M(1-q_0+\delta)$. By Lemma 1, the number of different frequency vectors we see at the output is upper bounded by

$$\begin{aligned}\mathcal{T}[M^\beta, M(1-q_0+\delta)] &= \binom{M^\beta+M(1-q_0+\delta)-1}{M(1-q_0+\delta)}\\ &= \binom{M^\beta+M(1-q_0+\delta)-1}{M^\beta-1}\\ &< \left(\frac{e(M^\beta+M(1-q_0+\delta))}{M^\beta}\right)^{M^\beta},\end{aligned}$$

where the second equality follows from $\binom{n}{k} = \binom{n}{n-k}$. Taking the logarithm we get

$$\begin{aligned}&\log\mathcal{T}[M^\beta, M(1-q_0+\delta)]\\ &\qquad\leq M^\beta((1-\beta)\log M+\log(1-q_0+\delta)+1).\end{aligned}$$

Dividing by $M^\beta L = M^\beta\beta\log(M)$ and letting $M \to \infty$ gives

$$\tilde{R} \leq (1-\beta)/\beta,$$

as desired. □

Acknowledgment

The authors would like thank Kannan Ramchandran and David Tse for helpful discussions in the early stages of this work. They would also like to thank Robert Grass for helpful discussions on modeling aspects of DNA storage channels.

References

[1] G. M. Church, Y. Gao, and S. Kosuri, "Next-generation digital information storage in DNA," *Science*, vol. 337, no. 6102, p. 1628, 2012.

[2] N. Goldman *et al.*, "Towards practical, high-capacity, low-maintenance information storage in synthesized DNA," *Nature*, vol. 494, no. 7435, pp. 77–80, Feb. 2013.

[3] R. N. Grass, R. Heckel, M. Puddu, D. Paunescu, and W. J. Stark, "Robust chemical preservation of digital information on DNA in silica with error-correcting codes," *Angew. Chem. Int. Ed.*, vol. 54, no. 8, pp. 2552–2555, Feb. 2015.

[4] H. T. Yazdi, Y. Yuan, J. Ma, H. Zhao, and O. Milenkovic, "A rewritable, random-access DNA-based storage system," *Sci. Rep.*, vol. 5, pp. 1–10, Sep. 2015.

[5] Y. Erlich and D. Zielinski, "DNA fountain enables a robust and efficient storage architecture," *Science*, vol. 355, no. 6328, pp. 950–954, Mar. 2017.

[6] L. Organick, S. D. Ang, Y.-J. Chen, R. Lopez, S. Yekhanin, K. Makarychev, M. Z. Racz, G. Kamath, P. Gopalan, and B. Nguyen, "Random access in large-scale DNA data storage," *Nature Biotechnol.*, vol. 36, no. 3, p. 242, 2018.

[7] P. L. Antkowiak *et al.*, "Low cost DNA data storage using photolithographic synthesis and advanced information reconstruction and error correction," *Nature Commun.*, vol. 11, no. 1, pp. 1–10, Dec. 2020.

[8] R. Heckel, G. Mikutis, and R. N. Grass, "A characterization of the DNA data storage channel," *Sci. Rep.*, vol. 9, no. 1, pp. 1–12, Dec. 2019.

[9] E. S. Lander and M. S. Waterman, "Genomic mapping by fingerprinting random clones: A mathematical analysis," *Genomics*, vol. 2, no. 3, pp. 231–239, Apr. 1988.

[10] A. S. Motahari, G. Bresler, and D. N. C. Tse, "Information theory of DNA shotgun sequencing," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6273–6289, Oct. 2013.

[11] M. S. Neiman, "Some fundamental issues of microminiaturization," *Radiotekhnika*, vol. 1, no. 1, pp. 3–12, 1964.

[12] E. Baum, "Building an associative memory vastly larger than the brain," *Science*, vol. 268, no. 5210, pp. 583–585, Apr. 1995.

[13] J. Sayir, "Codes for efficient data storage on DNA molecules," in *Proc. Talk Inform., Inference, Energy Symp.*, Cambridge, U.K., Mar. 2016. [Online]. Available: https://www.youtube.com/watch?v=TIlqFH0tvfQ&t=1437s

[14] J. Bornholt, R. Lopez, D. M. Carmean, L. Ceze, G. Seelig, and K. Strauss, "A DNA-based archival storage system," in *Proc. 21st Int. Conf. Architectural Support Program. Lang. Operating Syst.* New York, NY, USA: ACM, Mar. 2016, pp. 637–649.

[15] H. M. Kiah, G. J. Puleo, and O. Milenkovic, "Codes for DNA sequence profiles," *IEEE Trans. Inf. Theory*, vol. 62, no. 6, pp. 3125–3146, Jun. 2016.

[16] R. Gabrys, H. M. Kiah, and O. Milenkovic, "Asymmetric Lee distance codes: New bounds and constructions," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Apr. 2015, pp. 1–5.

[17] F. Sala, R. Gabrys, C. Schoeny, and L. Dolecek, "Exact reconstruction from insertions in synchronization codes," *IEEE Trans. Inf. Theory*, vol. 63, no. 4, pp. 2428–2445, Apr. 2017.

[18] A. Lenz, P. H. Siegel, A. Wachter-Zeh, and E. Yaakobi, "An upper bound on the capacity of the DNA storage channel," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Aug. 2019, pp. 1–5.

[19] A. Lenz, P. H. Siegel, A. Wachter-Zeh, and E. Yaakohi, "Achieving the capacity of the DNA storage channel," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 8846–8850.

[20] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 2012.

[21] M. Mitzenmacher, "On the theory and practice of data recovery with multiple versions," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2006, pp. 982–986.

[22] R. Heckel, I. Shomorony, K. Ramchandran, and D. N. C. Tse, "Fundamental limits of DNA storage systems," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017, pp. 3130–3134.

Ilan Shomorony (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Cornell University in 2014. He was a Post-Doctoral Scholar with UC Berkeley through the NSF Center for Science of Information (CSoI) till 2017. After that, he spent a year working as a Researcher and a Data Scientist with Human Longevity Inc., a personal genomics company. He is currently an Assistant Professor of electrical and computer engineering with the University of Illinois at Urbana–Champaign (UIUC), where he is a member of the Coordinated Science Laboratory. His research interests include information theory, communications, and computational biology. He received the NSF CAREER Award in 2021.

Reinhard Heckel (Member, IEEE) received the Ph.D. degree in electrical engineering from ETH Zurich. He was a Visiting Ph.D. Student with the Department of Statistics, Stanford University. He is currently a Rudolf Moessbauer Assistant Professor with the Department of Electrical and Computer Engineering (ECE), Technical University of Munich, and an Adjunct Assistant Professor with the Department of Electrical and Computer Engineering (ECE), Rice University, where he was an Assistant Professor, from 2017 to 2019. Before that, he was a Post-Doctoral Scholar with UC Berkeley—sharing an office with Ilan Shomorony—and a Researcher with the Cognitive Computing and Computational Sciences Department, IBM Research Zurich.. He is working in the intersection of machine learning and signal/information processing with a current focus on deep networks for solving inverse problems, learning from few and noisy samples, and DNA data storage.