Coded Shotgun Sequencing

Aditya Narayan Ravi, Alireza Vahid, *Senior Member, IEEE*, and Ilan Shomorony, *Member, IEEE*

***Abstract*—Most DNA sequencing technologies are based on the shotgun paradigm: many short reads are obtained from random unknown locations in the DNA sequence. A fundamental question, studied in Motahari *et al.*, (2013), is what read length and coverage depth (i.e., the total number of reads) are needed to guarantee reliable sequence reconstruction. Motivated by DNA-based storage, we study the coded version of this problem; i.e., the scenario where the DNA molecule being sequenced is a codeword from a predefined codebook. Our main result is an exact characterization of the capacity of the resulting *shotgun sequencing channel* as a function of the read length and coverage depth. In particular, our results imply that, while in the uncoded case, $O(n)$ reads of length greater than $2\log n$ are needed for reliable reconstruction of a length-n binary sequence, in the coded case, only $O(n/\log n)$ reads of length greater than $\log n$ are needed for the capacity to be arbitrarily close to 1.**

***Index Terms*—Shotgun sequencing, DNA storage, channel capacity, data storage, DNA sequencing.**

I. Introduction

OVER the last decade, advances in DNA sequencing technologies have driven down the time and cost of acquiring biological data tremendously. At the heart of this sequencing revolution was the development of *high-throughput shotgun sequencing* platforms. Rather than attempting to read a long DNA molecule from beginning to end, these platforms extract a large number of short reads from random locations of the target DNA sequence (e.g., the genome of an organism), in a massively parallel fashion. Sequencing must then be followed by an *assembly* step, where the reads are merged together based on regions of overlap with the intention of reconstructing the original DNA sequence.

In the context of this shotgun sequencing pipeline, it is natural to ask when it is possible, from an information-theoretic standpoint, to reconstruct a sequence from a random set of its substrings. More precisely, suppose we observe K random reads (i.e., substrings) of length L from an unknown length-n sequence x^n. What conditions on x^n, K and L guarantee that x^n can be reliably reconstructed from the observed reads? This problem was first studied from an information-theoretic point of view by Motahari *et al.* [1]. The authors considered the asymptotic regime where $n \to \infty$ and L scales as

$$L = \bar{L}\log n, \tag{1}$$

for a constant $\bar{L}$. They also defined $c = \frac{KL}{n}$ to be the *coverage depth*; i.e., the average number of times each symbol in x^n is sequenced. This appropriate scaling of the read length allowed the authors of [1] to show a surprising critical phenomenon: if x^n is an i.i.d. Ber(1/2) sequence, when $\bar{L} < 2$, reconstruction is impossible for any coverage depth c, but if $\bar{L} > 2$, reconstruction is possible as long as the coverage depth is at least the Lander-Waterman coverage $c_{LW} = \ln(n/\epsilon)$.

The Lander-Waterman coverage [2] is the minimum coverage needed to guarantee that all symbols in x^n are sequenced at least once with probability $1-\epsilon$. The result in [1] established a *feasibility region* for the shotgun sequencing problem, illustrated in blue in Figure 1. Notice that the number of reads required is linear in n since

$$K = \frac{n}{L}\cdot c_{LW} = \frac{n}{L}\ln\left(\frac{n}{\epsilon}\right) = \frac{n\ln(n/\epsilon)}{\bar{L}\log n} = \Theta(n).$$

One key aspect about the framework studied in [1] is that the sequence x^n is chosen "by nature" (which can be modeled as a random process as in [1] or as an unknown deterministic sequence as later done in [3], [4]). However, in recent years, significant advances in DNA synthesis technologies have enabled the idea of storing data in DNA, and several groups demonstrated working DNA-based storage systems [5]–[11]. In these systems, information was encoded into DNA molecules via state-of-the-art synthesis techniques, and later retrieved via sequencing. This emerging technology motivates the following question: How do the fundamental

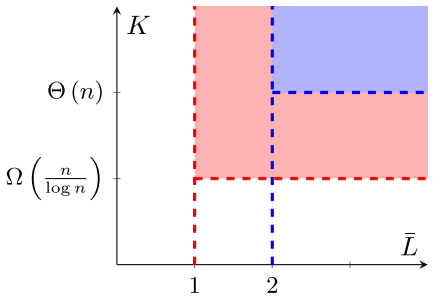

Fig. 1. The blue region describes a feasible region where the normalized read length $\bar{L}$ and the number of reads K satisfies conditions needed for perfect sequence reconstruction in the uncoded setting [1]. In the coded setting studied in this paper, the requirements for the capacity to be 1 are significantly less stringent: $\bar{L} > 1$ and K growing faster than $n/\log n$ suffices.

Manuscript received September 15, 2021; revised January 23, 2022; accepted February 7, 2022. Date of publication February 17, 2022; date of current version April 12, 2022. The work of Aditya Narayan Ravi and Ilan Shomorony was supported in part by the NSF under Grant CCF-2007597, and in part by the NSF CAREER Award under Grant CCF-2046991. The work of Alireza Vahid was supported in part by the NSF under Grant ECCS-2030285 and Grant CNS-2106692. *(Corresponding author: Aditya Narayan Ravi.)*

Aditya Narayan Ravi and Ilan Shomorony are with the Electrical and Computer Engineering Department, University of Illinois at Urbana–Champaign, Champaign, IL 61801 USA (e-mail: anravi2@illinois.edu; ilans@illinois.edu).

Alireza Vahid is with the Electrical Engineering Department, University of Colorado Denver, Denver, CO 80204 USA (e-mail: alireza.vahid@ucdenver.edu).

Digital Object Identifier 10.1109/JSAIT.2022.3151737

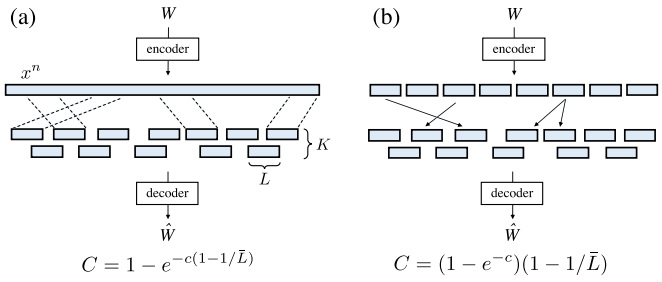

Fig. 2. Comparison between the (a) Shotgun Sequencing Channel (SSC) and the (b) Shuffling-Sampling channel from [12] and the corresponding capacity expressions. The input to the SSC is a single (binary) string x^n and the output are K random substrings of length L. In the Shuffling-Sampling channel, the input are M strings of length L, which are sampled with replacement to produce the channel output. Both capacity expressions can be written in terms of the expected coverage depth c and the normalized read length $\bar{L}$.

limits of shotgun sequencing from [1] change in the coded setting where x^n is chosen from a codebook?

Motivated by this question, in this paper we introduce the Shotgun Sequencing Channel (SSC). As illustrated in Figure 2(a), the channel input is a (binary) length-n sequence x^n, and the channel output are K random reads of length L from x^n. Each read is assumed to be drawn independently and uniformly at random from x^n and we consider the read length scaling in (1). Notice that this is essentially the same setup as in [1], except that the "genome" x^n is chosen from a codebook rather than decided by nature. Our goal is to characterize the capacity of this channel.

In order to build intuition it is worth considering the related setting of the *shuffling-sampling channel* [12], illustrated in Figure 2(b). In this case the input are M strings of length L, and the output are K strings, each chosen uniformly at random from the set of input strings. If we define the coverage depth for this setting as $c = \frac{KL}{ML} = K/M$, the result in [12] implies that, for $\bar{L} > 1$, the capacity of this channel is

$$C_{\text{shuf}} = \big(1 - e^{-c}\big)\big(1 - 1/\bar{L}\big), \tag{2}$$

and $C_{\text{shuf}} = 0$ for $\bar{L} \leq 1$. The term $(1 - e^{-c})$ captures the loss due to unseen input strings and $(1 - 1/\bar{L})$ captures the loss due to the unordered nature of the output strings (which becomes more severe the shorter the strings are).

Intuitively, the capacity of the SSC should depend on c and $\bar{L}$ in a similar way as in (2). The expected fraction of symbols in x^n that are read at least once can be shown to be $1 - e^{-c}$, which provides an upper bound to the capacity of the SSC. But it is not clear a priori which of the channels in Figure 2 should have the larger capacity. Our main result establishes that, for $\bar{L} \geq 1$, the capacity of the SSC is given by

$$C_{\text{SSC}} = 1 - e^{-c\left(1 - \frac{1}{\bar{L}}\right)}. \tag{3}$$

Notice that the dependence on $\bar{L}$ appears as the term $(1 - 1/\bar{L})$ in the exponent and, as $c \to \infty$, $C_{\text{SSC}} \to 1$ for any $\bar{L} > 1$. This is in contrast to the shuffling-sampling channel, where $C_{\text{shuf}} \to 1 - 1/\bar{L}$ as we increase the coverage depth c to infinity. Therefore, even in the high coverage depth regime, if $\bar{L} \approx 1$, $C_{\text{shuf}} \approx 0$. Furthermore, it can be verified that $C_{\text{shuf}} < C_{\text{SSC}}$ for any c and $\bar{L}$, establishing the advantage (from a capacity standpoint) of storing data on a long molecule of DNA as opposed to on many short molecules.

The above result also allows for an interesting comparison with the uncoded setting (i.e., the genome sequencing problem) of [1]. When we allow coding over the string, the critical threshold on the read length reduces to $\bar{L} > 1$, compared to $\bar{L} > 2$ for the uncoded setting. Moreover, in the SSC it is possible to achieve a capacity close to 1 by having the coverage depth be a large constant, while in the uncoded case the c needs to grow as $\log n$.

Background and Related Work: The first prototypes of DNA storage systems were presented in 2012 and 2013, when groups lead by Church *et al.* [5] and Goldman *et al.* [6] independently stored about a megabyte of data in DNA. In 2015, Grass *et al.* [7] demonstrated that millenia long storage times are possible by protecting the data using error-correcting codes. Yazdi *et al.* [8] showed how to selectively access parts of the stored data, and in 2017, Erlich and Zielinski [9] demonstrated that practical DNA storage can achieve very high information densities. In 2018, Organick *et al.* [10] scaled up these techniques and stored about 200 megabytes of data. We point out that, in all of these prototypes, data is stored on many short DNA molecules, as opposed to storing it in a single very-long DNA molecule. This is because synthesizing long strands of DNA is prohibitively expensive with current technology. Hence, this work seeks to answer what storage rates could be achieved if we were able to synthesize long DNA molecules at reasonable costs.

The prospect of practical DNA-based storage has motivated a significant amount of research into its theoretical underpinnings. The idea of coding over a set of strings that are shuffled and sampled was studied in several settings [12]–[18]. Many works have focused on developing explicit codes tailored to specific aspects of DNA storage. These include DNA synthesis constraints such as sequence composition [8], [9], [19], the asymmetric nature of the DNA sequencing error channel [20], the need for codes that correct insertion errors [21], and the need for techniques to allow random access [8].

The problem of reconstructing a string from a set of its subsequences has also been considered in various settings. Several works studied the problem of genome sequencing and assembly from an information-theoretic standpoint [1], [3], [4], [22]. The trace reconstruction problem is another related setting where one observes (non-contiguous) subsequences of the input sequence and attempt to reconstruct it [23]–[25].

A very relevant related setting is the problem of reconstructing a string from its substring spectrum [26], [27]. Our setting is similar to this problem in two ways: (i) that both problems look at trying to reconstruct strings from substrings of fixed lengths, in general with overlaps, and (ii) the string is chosen from a codebook. However, these works have focused on the setting where a noisy substring spectrum (the multiset of all substrings) is available, while we consider that a fixed number of reads (or substrings) are extracted from random locations. Moreover, these works proposed explicit code constructions, while we focus on the problem of capacity characterization.

II. Problem Setting

We consider the Shotgun Sequencing Channel (SSC), shown in Figure 2(a). The transmitter sends a length-n binary string $X^n \in \{0,1\}^n$, corresponding to a message $W \in [1 : 2^{nR}]$. The channel output is a set of length-L binary strings $\mathcal{Y}$. The channel chooses K starting points uniformly at random, represented by the random vector $T^K \in [1 : n]^K$. The vector T^K is assumed to be sorted in a non-decreasing order. Length-L reads are then sampled with T_i, $i = 1, \ldots, K$ as their starting points. We allow the reads to "wrap around" X^n; i.e., if for any i, $T_i + L > n$, we concatenate bits from the start of X^n to form length-L reads. For example if $T_i = n - 2$ and $L = 5$, then the read $\vec{Y}$ associated with this starting location is $\vec{Y} = [X_{n-2}, X_{n-1}, X_n, X_1, X_2]$. Notice that the channel effectively treats the codeword as circular, equivalent to the circular DNA model considered in [1]. The unordered multiset $\mathcal{Y} = \{\vec{Y}_1, \vec{Y}_2, \ldots, \vec{Y}_K\}$ of reads resulting from this sampling process is the channel output.

The expected number of times a symbol from X^n is sequenced is defined as the coverage depth c, given by

$$c \coloneqq \frac{KL}{n}.$$

We focus on the regime where the length of the reads sampled is much smaller than the block length n. In particular, as shown in previous works [12], [28]–[31], the regime $L = \Theta(\log n)$ is of interest from a capacity standpoint. Hence, as in [1], we fix a normalized length $\bar{L}$ and define $L \coloneqq \bar{L} \log n$. Notice that, in this regime, the total number of reads is

$$K = \frac{cn}{\bar{L} \log n} = \Theta\left(\frac{n}{\log n}\right),$$

which is a $\log n$ factor smaller than what is needed in the uncoded setting from [1].

We define an achievable rate in the usual way. More precisely, a $(2^{nR}, n)$-code consists of a message set $[1 : 2^{nR}]$, an encoder that assigns codeword $x^n(W)$ to a $W \in [1 : 2^{nR}]$, and a decoder that assigns an estimate $\hat{W}(\mathcal{Y}) \in [1 : 2^{nR}]$. Rate R is achievable if there exists a sequence of $(2^{nR}, n)$ codes whose error probability tends to zero as $n \to \infty$. The capacity C of the SSC is the supremum over achievable rates.

Notation: $\log(\cdot)$ represents the logarithm in base 2. For functions $a(n)$ and $b(n)$, we say $a(n) = o(b(n))$ or $b(n) = \Omega(a(n))$ if $a(n)/b(n) \to 0$ as $n \to \infty$. Further, we say that a function $a(n) = \Theta(f(n))$ if there exist $n_0 \in \mathbb{N}, k_1, k_2 \in (0, \infty)$, such that $k_1 f(n) \leq a(n) \leq k_2 f(n)$ $\forall n \geq n_0$. For an event A, we let $\mathbf{1}_A$ be the binary indicator of A. For a set B, $|B|$ indicates the cardinality of that set and B^c denotes the complement of that set.

III. Main Results

The DNA storage problem considered here has two important properties: (i) the reads in general overlap with each other and (ii) the set of reads is unordered. Property (i) was explored in the context of genome sequencing [1]. Intuitively, the overlaps between the reads allow them to be merged in order to reconstruct longer substrings of X^n. Property (ii) has been analyzed before in the context of several works on DNA storage. In particular, in the context of the shuffling-sampling channel from [12], illustrated in Figure 2(b), the input to the channel is a set of strings of length L, and the capacity is given by $C_{\text{shuf}} = (1 - e^{-c})(1 - 1/\bar{L})$.

Notice that, in the case of the shuffling-sampling channel, the output strings have no overlaps (they can only be non-overlapping or identical). In the context of the SSC, on the other hand, overlaps can provide useful information to fight the lack of ordering of the output strings. Our main result captures the capacity gains achieved by optimally exploiting the overlaps. Specifically, we characterize the capacity of the SSC for any coverage depth c and normalized read length $\bar{L}$.

Theorem 1: For any $c > 0$ and $\bar{L} > 0$, the capacity of the Shotgun Sequencing Channel is

$$C = \left(1 - e^{-c(1-1/\bar{L})}\right)^+. \tag{4}$$

In order to prove Theorem 1, we consider a random coding argument and develop a careful decoding algorithm that allows for a tight analysis of the error probability. For the converse we use a novel constrained-genie argument, which specifically tackles property (i).

Notice that the capacity of the SSC given in Theorem 1 is zero when $\bar{L} \leq 1$. An intuitive reason for this is that when $\bar{L} < 1$, the number of possible distinct length-L sequences is just $2^{\bar{L} \log n} = n^{\bar{L}} = o(n/\log n) = o(K)$, and many reads must be identical. This can be used to show that the decoder cannot extract any meaningful information from $\mathcal{Y}$. Section V discusses this further. When $\bar{L} = 1$, this same intuition doesn't hold, but as a consequence of the continuity of C, we have $C = 0$ when $\bar{L} = 1$. This is indeed true as seen in Section V.

In order to interpret the capacity expression in (4) notice that the probability that a given symbol in X^n is not sequenced by any of the K reads is

$$(1 - L/n)^K = (1 - L/n)^{\frac{cn}{L}} \to e^{-c}, \tag{5}$$

as $n \to \infty$. Hence the expected fraction of symbols in X^n that are covered by at least one read is asymptotically close to $1 - e^{-c}$. If instead of reads of length $L = \bar{L} \log n$ we had reads of length $(\bar{L} - 1) \log n$, the coverage depth would be

$$c' = \frac{K(\bar{L} - 1) \log n}{n} = c(1 - 1/\bar{L}),$$

and the expected fraction of symbols in X^n that would be sequenced would be $1 - e^{-c'} = 1 - e^{-c(1-1/\bar{L})}$. Hence, the capacity expression in Theorem 1 suggests that, on average, $\log n$ bits from each read are used for ordering information, while the remaining $(\bar{L} - 1) \log n$ bits provide data information.

It is also interesting to compare the capacity of the SSC and the capacity of the shuffling-sampling channel $C_{\text{shuf}} = (1 - e^{-c})(1 - 1/\bar{L})$. Note from Figure 3, that C_{shuf} is strictly upper bounded by (4). This shows that given a coverage depth c, there are significant gains in terms of capacity to be obtained if we store data on a long DNA molecules instead of many short DNA molecules. Moreover if we let the coverage depth $c \to \infty$; i.e., allow for a large number of samples, when $\bar{L} > 1$, $C_{\text{shuf}} \to 1 - 1/\bar{L}$, while $C \to 1$. In particular, when reads are very short and $\bar{L} \approx 1$, $C_{\text{shuf}} \approx 0$, while the capacity of the SSC can be close to 1 for large enough c.

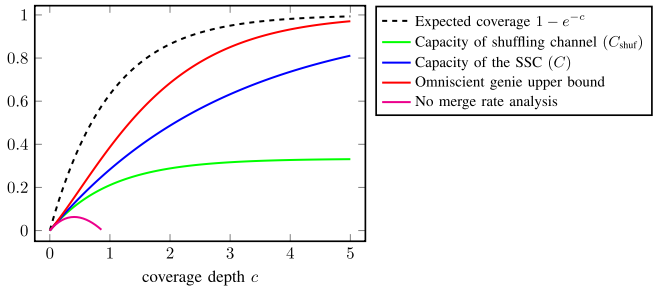

Fig. 3. Comparison between the capacity of the SSC $C = 1 - e^{-c(1-\frac{1}{\bar{L}})}$ with $\bar{L} = 1.5$, the capacity of the shuffling channel with fragments of fixed length L, the maximum rates achieved on the SSC when we allow a genie to merge the reads and the maximum rate achieved if reads are not merged.

It is also interesting to compare Theorem 1 to the uncoded setting of genome sequencing studied in [1]. As discussed in Section I and illustrated in Figure 1, the results in [1] show that the perfect reconstruction reconstruction (with error asymptotically going to 0) of a random (uncoded) string X^n, can be done as long as $\bar{L} > 2$ and $K = \Theta(n)$. In contrast, for the coding setting of the SSC, as long as $\bar{L} > 1$ and $K = \Theta(\frac{n}{\log n})$, we can obtain a positive capacity. Moreover as discussed in the introduction, since as $c \to \infty$, $C = 1$ we can claim that if $K = \Omega(n/\log n)$, then for $\bar{L} > 1$, $C = 1$. This means that coding allows us to considerably reduce the threshold on sampled read size (by a factor of half) and the number of samples (by a factor of nearly $1/\log n$), while still admitting asymptotically perfect reconstruction.

The remainder of the paper is organized as follows. In Sections IV and V, we prove the achievability and converse of Theorem 1 respectively. We conclude the paper in Section VI.

IV. Achievability

We use a random coding argument to prove the achievability of Theorem 1. We generate a codebook with 2^{nR} codewords of length n, independently picking each letter Ber(1/2). Let the codebook be $\mathcal{C} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{2^{nR}}\}$. The encoder chooses the codeword corresponding to the message $W \in [1 : 2^{nR}]$, and sends $\mathbf{x}_W$ across the Shotgun Sequencing Channel. The output, $\mathcal{Y}$, is presented to the decoder. For the analysis and without loss of generality, we assume $W = 1$.

The optimal decoder looks for a codeword that contains all the reads in $\mathcal{Y}$ as substrings. Analyzing the error probability of this optimal decoder, however, is hard. We therefore aim to develop a decoding rule that is simple enough to analyze.

A. Analysis Without Exploiting Overlaps

The fact that, in general, there are overlaps between the reads is an important feature of the output, since they allow reads to be merged, and this should be taken into account in the decoding rule. To motivate this, let us first bound the error probability without exploiting the overlaps for merging reads.

We say that the ith bit of X^n is *covered* if there is a read with starting position in $\{i - L + 1, i - L + 2, \ldots, i\}$, where the indices wrap around X^n. We then define the *coverage* as $\Phi = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{\{i\text{th bit is covered}\}}$.

Lemma 1 (Coverage): For any $\epsilon > 0$, as $n \to \infty$,

$$\Pr\left(\left|\Phi - \left(1 - e^{-c}\right)\right| > \epsilon\left(1 - e^{-c}\right)\right) \to 0. \tag{6}$$

The proof of this lemma is presented in Appendix A. Note that $\lim_{n\to\infty} E[\Phi] = 1 - e^{-c}$ as described in (5), and thus, Lemma 1 guarantees that the coverage Φ is concentrated around its expected value. We use this fact to discern how many bits in a candidate codeword need to match the bits sampled in the reads.

The decoding rule we consider is as follows: The decoder looks for the codeword in the codebook that contains all reads as substrings of that codeword and that for an $\epsilon > 0$, the coverage of these reads $> (1-\epsilon)(1-e^{-c})$. It declares an error if more than one such codeword exists. We want to bound the probability of error $\mathcal{E}$ based on Lemma 1. We define $B := (1-\epsilon)(1-e^{-c})$ and follow steps similar to [29] to obtain

$$\begin{aligned}\Pr(\mathcal{E}) &= \Pr(\mathcal{E}|W=1) \le \Pr(\mathcal{E}|W=1, \Phi \ge B) + \Pr(\Phi < B)\\ &\overset{(a)}{\le} 2^{nR} \times n^K \times \frac{1}{2^{nB}} + o(1) = 2^{n(R-(K\log n)/n-B)} + o(1),\end{aligned} \tag{7}$$

where (a) holds because there are at most n^K ways to arrange the K reads on a codeword and, given an arrangement, at least nB bits of an incorrect codeword would need to match our reads to create an error. Since $(K\log n)/n = c/\bar{L}$, in order for $\Pr(\mathcal{E}) \to 0$ as $n \to \infty$, we would need

$$R \le 1 - e^{-c} - c/\bar{L}. \tag{8}$$

The achievable rate obtained from this analysis is plotted in Figure 3 in magenta. This rate is suboptimal and in fact (above a critical value of c) reduces as the coverage depth increases. This arises because when we bound the number of ways to arrange the reads on a length-n codeword by n^K, it does not take into account overlaps between the reads. To be able to discern higher rates, we need to develop a way to utilize the fact that, in general, many of the reads overlap with each other.

B. Using Overlaps to Merge Reads

The analysis above indicates the need to merge the reads before we compare them to candidate codewords. Unfortunately, merging reads is not a straightforward process because reads $\vec{Y}_i$ and $\vec{Y}_j$ may have an overlap even if they do not correspond to overlapping segments of X^n. In general, the merging process will be prone to errors and we need to develop a decoding algorithm that considers merges in a careful way.

In Section II, we defined the unknown vector T^K to be the ordered starting positions of the reads in $\mathcal{Y}$. Thus, without loss of generality we assume that $\vec{Y}_i$ starts at T_i. We define the *successor* of $\vec{Y}_i$ as $\vec{Y}_{i+1}$. We assume Y_1 is the successor of Y_K. Now we need a consistent definition to characterize how large the overlap of a given read is.

Definition 1 (Overlap Size): The overlap size of a read is defined as the number of bits the suffix of the read shares with its successor. It has an overlap size of 0 if no bits are shared (i.e., if a read and its successor have no overlap).

The above definition implies that the overlap size of $\vec{Y}_i$ is $(L-(T_{i+1}-T_i))^+$. Notice that some reads might share some of

their prefix bits with a predecessor read, but we do not consider this a contribution to the overlap size of that read. Intuitively speaking, since each bit of X^n was generated independently as a Ber(1/2) random variable, we would expect larger overlap sizes to be easily discerned as compared to smaller ones. Therefore, we would need to know: (a) how many pieces exist of particular overlap sizes, and (b) given an overlap size, how "easy" it is to merge a read with its successor.

To handle (a), we define $G(\gamma)$ as a random variable that counts the number of reads with overlap size $\gamma \log n$, where $\gamma \in \Gamma := \{\frac{1}{\log n}, \frac{2}{\log n}, \ldots, \bar{L}\}$. Thus, γ is chosen from a finite set that depends on n. If $G(\gamma)_i = \mathbf{1}_{\{\vec{Y}_i \text{ has an overlap size of } \gamma \log n\}}$, $G(\gamma) := \sum_{i=1}^{K} G(\gamma)_i$.

To capture (b), given a binary string $\vec{z}$, we define the random variable $M_{\vec{z}}$ as the number of times $\vec{z}$ appears as the prefix of a read in $\mathcal{Y}$. Note that the length of $\vec{z}$ is in $[1 : \bar{L} \log n]$. Let $\mathcal{Z}$ be the set of all binary strings with lengths in $[1 : \bar{L} \log n]$. If we can identify the merges correctly, we are left with a set of variable-length strings called islands, formally defined next.

Definition 2 (Islands): The set of non-overlapping substrings that are obtained after merging all the reads to their successors based on positive overlap sizes are called islands.

Let K' be the number of islands. Then, we have:

Lemma 2 (Number of Islands): For any $\epsilon > 0$, as $n \to \infty$,

$$\Pr\left(\left|K' - Ke^{-c}\right| \geq \epsilon K e^{-c}\right) \to 0. \tag{9}$$

The proof of this lemma is available in Appendix B. Similar to the previous lemma, Lemma 2 guarantees that the number of islands K' is concentrated around its expectation Ke^{-c}. Lemmas 1 and 2 are used in the later part of the decoding to look at different arrangements of the non-overlapping islands and the bits that match these arrangements. However, to use these results, the decoder would first need to obtain the non-overlapping islands (the decoder only has the reads currently). The following lemmas give the decoder some guidelines on how to construct these islands from the reads.

Lemma 3 (Number of Potential Overlaps): For any $\epsilon > 0$,

$$\Pr\left(\bigcup_{\vec{z}\in\mathcal{Z}:\gamma(\vec{z})\leq 1-\epsilon} \left\{\left|M_{\vec{z}} - Kn^{-\gamma(\vec{z})}\right| \geq \epsilon K n^{-\gamma(\vec{z})}\right\}\right) \to 0 \text{ and}$$

$$\Pr\left(\bigcup_{\vec{z}\in\mathcal{Z}:\gamma(\vec{z})>1-\epsilon} \left\{M_{\vec{z}} \geq n^{\epsilon}\right\}\right) \to 0, \tag{10}$$

as $n \to \infty$, where we define $\gamma(\vec{z}) := |\vec{z}|/\log n$.

Lemma 3 considers two separate cases for binary strings based on their length. For strings $\vec{z}$ with length at most $(1 - \epsilon)\log n$, Lemma 3 states that $M_{\vec{z}}$ is close to its mean

$$Kn^{-\gamma(\vec{z})} = cn^{1-\gamma(\vec{z})}/\left(\bar{L}\log n\right).$$

For strings $\vec{z}$ with length greater than $(1 - \epsilon)\log n$, the same concentration result does not hold, and Lemma 3 simply states that $M_{\vec{z}} < n^{\epsilon}$ with high probability.

Lemma 4 (Number of Reads of a Given Overlap Size):

$$\Pr\left(\bigcup_{\gamma\in\Gamma}\left\{|G(\gamma) - \bar{G}(\gamma)| \geq \epsilon \bar{G}(\gamma)\right\}\right) \to 0 \tag{11}$$

as $n \to \infty$, for all $\epsilon > 0$, where $\bar{G}(\gamma) := E[G(\gamma)]$.

Lemma 4 give us a handle on the expected number of overlaps of each size, which will be used by the decoder when trying to construct the islands from the reads. Lemmas 3 and 4 are proved in Appendices C and D.

The decoding procedure starts with a brute-force search over ways to merge the reads into islands, which we refer to as the Partition and Merge (PM) algorithm. We will first explain it in words and follow it by outlining the exact algorithm. First, the decoder considers all possible partitions of the reads into L groups, by assigning potential overlap sizes to each read. This can be done by looking at all ways of assigning a number in $[0 : L]$ to each of the reads. To make this precise, we can look at all possible vectors of the form $\vec{p} := (p_1, p_2, \ldots, p_K) \in [0 : L]^K$ and call them *partition vectors*. Each element p_i of the vector corresponds to an assigned overlap size of read $\vec{Y}_{\sigma(i)}$ for some permutation σ of the elements of $\mathcal{Y}$. Thus, each partition vector along with a permutation σ can be viewed as assigning an overlap size to each read. It is easy to see the total number of such partition vectors (and hence the total possible partitions) will be $P := (L + 1)^K$.

Rather than considering all P partitions, we will only consider partitions that satisfy the bounds implied by Lemmas 1–4. To make this requirement precise, we define for a partition vector $\vec{p}$, $G(\vec{p}, \gamma)$ to be the number of reads in $\mathcal{Y}$ that would have an overlap size of $\gamma \log n$ according to partition vector $\vec{p}$, which can be written as

$$G(\vec{p}, \gamma) = |\{i : p_i = \gamma \log n\}|. \tag{12}$$

Note that since the number of potential islands is exactly equal to the number of reads with overlap size zero, the total number of islands according to $\vec{p}$ is $G(\vec{p}, 0)$. Moreover we define $\Phi(\vec{p})$ as the total coverage of the reads according to $\vec{p}$, given by $\Phi(\vec{p}) := KL - \sum_{i=1}^{K} p_i$. We then define $\mathcal{P}$ as the set of all $\vec{p}$ such that (for fixed $\epsilon > 0$):

- $|\Phi(\vec{p}) - (1 - e^{-c})| \leq \epsilon(1 - e^{-c})$ (i.e., coverage is close to expected coverage),
- $|G(\vec{p}, 0) - Ke^{-c}| \leq \epsilon K e^{-c}$ (i.e., number of islands is close to expected number of islands),
- $|G(\vec{p}, \gamma) - \bar{G}(\gamma)| \leq \epsilon \bar{G}(\gamma)$ for all $\gamma \in \Gamma$ (i.e., number of reads with overlap size $\gamma \log n$ is close to its expectation).

Therefore, $\mathcal{P}$ restricts the total number of partition vectors to a smaller set of partition vectors that are admissible according to Lemmas 1, 2 and 4.

Now, for each partition vector $\vec{p} \in \mathcal{P}$, we take all possible $K!$ permutations σ of the reads. For each permutation, all of the reads are compared to their successors. If every read can be successfully merged with its successor with the assigned overlap size, we retain the set of substrings formed after these merges as a Candidate Island set, and add it to the set CI. Notice that CI is a set of sets of variable-length strings. This procedure is summarized in Figure 4 and Algorithm 1. After completion of Algorithm 1, the decoder checks, for each set of candidate islands in CI, whether there exists a codeword that contains all the candidate islands as substrings. If only one such codeword is found, the decoder outputs its index. Otherwise, an error is declared.

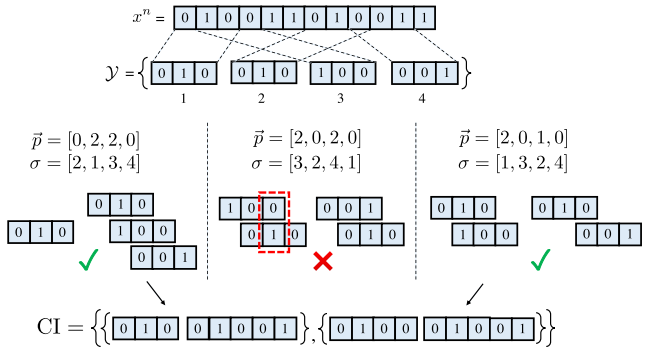

Fig. 4. The decoder receives the shotgun sequenced reads $\mathcal{Y}$ and performs the Partition and Merge procedure. For each partition $\vec{p} \in [0:L]^K$ of the K reads according to overlap size, and each ordering of the reads σ, the decoder attempts to merge the reads into islands based on σ and $\vec{p}$. The figure shows this procedure for three choices of $\vec{p}$ (out of $(1+L)^K$) and three choices of σ (out of $K!$). If the merging of all reads is successful for some p and σ, the set of resulting islands is added to the set CI.

Algorithm 1: Partition and Merge

for *each partition vector* $\vec{p} \in \mathcal{P}$ **do**
  for *each permutation* σ *of* $[1:K]$ **do**
    check if suffix of length p_i of $\vec{Y}_{\sigma(i)}$ matches prefix of $\vec{Y}_{\sigma(i+1)}$, for $i = 1, \ldots, K$
    if *prefix and suffix match for* $i = 1, \ldots, K$ **then**
      Merge reads according to overlaps
      Add set of resulting islands to CI
return CI

Let the event that the decoder makes an error be $\mathcal{E}$. An error occurs if more than one codeword contains any of the CI sets as substrings. We define $B_1 := (1+\epsilon)Ke^{-c}$, $B_2 := (1-\epsilon)(1-e^{-c})$, $B_3(\gamma) := (1+\epsilon)n^{1-\gamma}$ for $\gamma \le 1-\epsilon$, $B_3(\gamma) = n^{\epsilon}$ for $\gamma > 1-\epsilon$ and $B_4(\gamma) := (1+\epsilon)\bar{G}(\gamma)$, and we define the corresponding undesired events as

$$\mathcal{B}_1 = \{K' > B_1\}, \quad \mathcal{B}_2 = \{\Phi < B_2\}$$
$$\mathcal{B}_3 = \bigcup_{\vec{z}\in\mathcal{Z}} \{M_{\vec{z}} > B_3(\gamma(\vec{z}))\}, \quad \mathcal{B}_4 = \bigcup_{\gamma\in\Gamma} \{G(\gamma) > B_4(\gamma)\}.$$

From Lemmas 1, 2, 3 and 4, if we let

$$\mathcal{B} = \mathcal{B}_1 \cup \mathcal{B}_2 \cup \mathcal{B}_3 \cup \mathcal{B}_4, \tag{13}$$

we have $\Pr(\mathcal{B}) \to 0$. Note that conditioned on $\bar{\mathcal{B}}$, we are guaranteed that the CI set outputs the true island set. This is because exactly one partition and one arrangement given that partition correspond to the true order in which the reads were sampled. Before we use this to bound the probability of error, notice that the error event depends on the total number of CI sets output by the PM algorithm. In general, this is not a deterministic value. We will define $\overline{CI}_n$ as an upper bound on the number of CI sets conditioned on $\bar{\mathcal{B}}$. We claim that conditioned on $\bar{\mathcal{B}}$, after the PM algorithm, the resulting CI (which is a set of sets of binary strings) satisfies

$$|\text{CI}| \le P \times \prod_{\gamma\le 1-\epsilon} B_3(\gamma)^{B_4(\gamma)} \times \prod_{\gamma>1-\epsilon} n^{\epsilon B_4(\gamma)} := \overline{CI}_n. \tag{14}$$

To see this, first we notice that $|\mathcal{P}| \le P$. According to a given partition vector $\vec{p} \in \mathcal{P}$, there are at most $B_4(\gamma)$ reads with overlap size $\gamma \log n$. Given a read $\vec{Y}_i$ with assigned overlap size $\gamma \log n$, when $\gamma \le 1-\epsilon$, there are at most $B_3(\gamma)$ reads whose prefix matches the $(\gamma \log n)$-suffix of $\vec{Y}_i$ and, therefore, at most $B_3(\gamma)$ potential valid merges. Therefore, for a given overlap size $\gamma \log n$, $\gamma \le 1$, there at most $B_3(\gamma)^{B_4(\gamma)}$ merge possibilities. However, when $\gamma > 1-\epsilon$, we know that for the given read, there at most n^{ϵ} potential valid merges. Therefore there are at most $n^{\epsilon B_4(\gamma)}$ merge possibilities. We thus bound the probability of error averaged over all codebooks as

$$\begin{aligned}\Pr(\mathcal{E}) &= \Pr(\mathcal{E}|W=1) \le \Pr\big(\mathcal{E}|W=1,\bar{\mathcal{B}}\big) + \Pr(\mathcal{B}) \\ &\overset{(a)}{\le} 2^{nR} \times \overline{CI}_n \times n^{B_1} \times \frac{1}{2^{nB_2}} + o(1) \\ &= 2^{nR+\log\overline{CI}_n + B_1\log n - nB_2} + o(1) \\ &= 2^{nR+\log\overline{CI}_n + (1+\epsilon)Ke^{-c}\log n - n(1-\epsilon)(1-e^{-c})} + o(1). \end{aligned} \tag{15}$$

This follows because an error occurs if any of the $2^{nR}-1$ codewords ($W \ne 1$) contain any of the sets of candidate islands in CI (which is upper bounded by $\overline{CI}_n$ when conditioned on $\bar{\mathcal{B}}$). Each of the sets in CI contains at most B_1 islands and a total island length of at least nB_2 bits. Hence, there are at most n^{B_1} ways to arrange the islands on a codeword and, given one such arrangement, an an erroneous codeword must match these islands at at least nB_2 bits.

Let us compare (7) and (15). The term n^K in (7), which bounds the number of arrangements of reads on a candidate codeword, is replaced by $n^{B_1} \times \overline{CI}_n$ in (15), which carefully takes into account both the cost of merging reads together and the number of arrangements after all the reads are merged together to form CIs. This improvement, as we see below, is crucial to achieve optimal rates.

In order to have $\Pr(\mathcal{E}) \to 0$ in (15), we require

$$\begin{aligned} R &\le \lim_{n\to\infty}\left((1-\epsilon)\big(1-e^{-c}\big) - (1+\epsilon)\frac{ce^{-c}}{\bar{L}} - \frac{1}{n}\log\overline{CI}_n\right) \\ &= (1-\epsilon)\big(1-e^{-c}\big) - (1+\epsilon)\frac{ce^{-c}}{\bar{L}} - \lim_{n\to\infty}\frac{1}{n}\log\overline{CI}_n \end{aligned}$$

The following lemma, proved in Appendix E, evaluates the last term in the above expression.

Lemma 5: The upper bound $\overline{CI}_n$ on |CI| satisfies

$$\lim_{n\to\infty}\frac{1}{n}\log\overline{CI}_n \le e^{-c\left(1-\frac{1}{\bar{L}}\right)} - \left(\frac{c}{\bar{L}}+1\right)e^{-c} + f(\epsilon),$$

where $f(\epsilon) \to 0$ as $\epsilon \to 0$.

This Lemma is proved in Appendix E. From Lemma 5 and by letting $\epsilon \to 0^+$, we conclude that all rates

$$R < 1 - e^{-c\left(1-\frac{1}{\bar{L}}\right)},$$

are achievable, completing the achievability of Theorem 1.

V. Converse

To prove the converse, we borrow some insights from the achievable scheme in Section IV. The idea is to have a genie-aided channel in which the reads are already merged into islands. However, this step needs to be carried out with care.

More specifically, an omniscient genie, which merges all correct overlaps, would be too powerful and therefore result in a loose upper bound. Instead, we use a constrained genie that can only merge reads with overlap sizes above a certain threshold, and the resulting upper bound matches achievable rates of Section IV.

The proof is organized as follows. We first look at the case when $\bar{L} \leq 1$ and show the capacity is zero. Next, we provide the argument with an omniscient genie, which as we mentioned, results in a loose upper bound. We then show how to carefully constrain this genie to obtain a tighter upper bound that matches the achievable rates.

Short reads ($\bar{L} < 1$): The intuition is that, when $\bar{L} < 1$, the number of possible distinct length-L sequences is just $2^{\bar{L}\log n} = n^{\bar{L}} = o(n/\log n) = o(K)$, and many reads must be identical.

By Fano's inequality, followed by the counting argument similar to [12], we have that

$$R \leq \frac{H(\mathcal{Y})}{n} \overset{(a)}{\leq} \frac{1}{n}\log\binom{K+2^L-1}{K} \leq \frac{K}{n}\log\left(\frac{K+2^L-1}{K}\right)$$
$$= \frac{K}{n}\log\left(1+\left(\frac{2^L-1}{K}\right)\right) \overset{(b)}{\leq} K\frac{2^L}{nK} = cn^{\bar{L}-1} \to 0,$$

as $n \to \infty$, when $\bar{L} < 1$. In the above set of equations (a) is due to Lemma 1 in [12] and (b) is because $\log(1+x) \leq x$, for $x > 0$. Note that this holds irrespective of the value of K.

Omniscient genie: As observed in Section IV, merging the reads correctly to form islands is helpful while decoding the message. To capture this in the converse, suppose we have a genie that merges the reads into islands a priori and thus, forms variable-length non-overlapping substrings. Let $\mathcal{Y}'$ be the multiset of the islands this genie creates from $\mathcal{Y}$. Then, from Fano's inequality,

$$R \leq \lim_{n\to\infty}\frac{1}{n}I\big(X^n;\mathcal{Y}\big) \leq \lim_{n\to\infty}\frac{H(\mathcal{Y})}{n}$$
$$\leq \lim_{n\to\infty}\frac{H\big(\mathcal{Y},\mathcal{Y}'\big)}{n} = \lim_{n\to\infty}\frac{H\big(\mathcal{Y}'\big)+H\big(\mathcal{Y}|\mathcal{Y}'\big)}{n}. \quad (16)$$

Consider the second term in the numerator of the above expression. Notice that this term intuitively looks at the uncertainty in the set of reads given the fully merged islands. It would be helpful to get a handle on the number of reads per island to bound this term. Thus we define D as the maximum number of reads making up an island.

Lemma 6 (Maximum Number of Reads Per Island): For any $\gamma_0 > -1/\log(1-e^{-c})$, as $n \to \infty$,

$$\Pr(D > \gamma_0 \log n) \to 0. \quad (17)$$

This lemma is proved in Appendix F. Now define $\mathcal{B}$ as in (13). For a fixed $\gamma_0 > -1/\log(1-e^{-c})$,

$$H(\mathcal{Y}|\mathcal{Y}') \leq H\Big(\mathcal{Y}, \mathbf{1}_{\bar{\mathcal{B}}, D\leq\gamma_0\log n}|\mathcal{Y}'\Big) \leq H\Big(\mathcal{Y}|\mathcal{Y}', \mathbf{1}_{\bar{\mathcal{B}}, D\leq\gamma_0\log n}\Big)$$
$$+\ 1 \leq 1 + H\big(\mathcal{Y}|\mathcal{Y}', \overline{\mathcal{B}}, D \leq \gamma_0\log n\big)\Pr\big(\overline{\mathcal{B}}, D \leq \gamma_0\log n\big)$$
$$+\ H\big(\mathcal{Y}|\mathcal{Y}', \{\mathcal{B} \text{ or } D > \gamma_0\log n\}\big))(\Pr(\mathcal{B} \text{ or } D > \gamma_0\log n))$$
$$\leq 1 + H\big(\mathcal{Y}|\mathcal{Y}', \overline{\mathcal{B}}, D \leq \gamma_0\log n\big)$$
$$+\ H(\mathcal{Y}|\{\mathcal{B} \text{ or } D > \gamma_0\log n\})(\Pr(\mathcal{B}) + \Pr(D > \gamma_0\log n)). \quad (18)$$

Now, since $\mathcal{Y}$ is fully determined by X^n and the read starting points T^K, we have that $H(\mathcal{Y}|\{\mathcal{B} \text{ or } D > \gamma_0\log n\}) \leq 2n$. We can thus claim that

$$\lim_{n\to\infty}\frac{1}{n}H(\mathcal{Y}|\{\mathcal{B} \text{ or } D > \gamma_0\log n\})(\Pr(\mathcal{B}) + \Pr(D > \gamma_0\log n))$$
$$\overset{(a)}{\leq} \lim_{n\to\infty} 2(\Pr(\mathcal{B}) + \Pr(D > \gamma_0\log n)) = 0, \quad (19)$$

where (a) is due to the fact that $\Pr(\mathcal{B}) \to 0$ and Lemma 6.

We now focus on the first entropy term in (18). Let Q be the total number of substrings of length-L in an island. Given the maximum number of reads per island $D \leq \gamma_0 \log n$, we can say that the total length of an island cannot exceed $L \times \gamma_0 \log n = \bar{L}\gamma_0 \log^2 n$. Therefore the total number of substrings per island Q cannot exceed

$$Q \leq \bar{L}\gamma_0 \log^2 n. \quad (20)$$

Since there are $K' \leq K$ islands in $\mathcal{Y}'$, we can say that the total number of substrings of length-L in $\mathcal{Y}'$ is upper bounded by $KQ \leq K \times \bar{L}\gamma_0 \log^2 n = \gamma_0 cn \log n$.

Now for the term $H(\mathcal{Y}|\mathcal{Y}', \overline{\mathcal{B}}, D \leq \gamma_0 \log n)$, $\mathcal{Y}$ can be thought of as an histogram over these KQ length-L substrings with the sum of the histogram entries being exactly K. Following the counting argument from [12, Lemma 1]

$$H\big(\mathcal{Y}|\mathcal{Y}', \overline{\mathcal{B}}, D \leq \gamma_0\log n\big) \leq \log\binom{KQ+K-1}{K}$$
$$\leq K\log\left(\frac{e(KQ+K-1)}{K}\right) \leq \frac{cn}{\log n}\log\Big(\big(ec\bar{L}\gamma_0\big)\log^2 n\Big).$$

This implies that

$$\lim_{n\to\infty}\frac{1}{n}\big(1 + H\big(\mathcal{Y}|\mathcal{Y}', \overline{\mathcal{B}}, D \leq \gamma_0\log n\big)\big) = 0. \quad (21)$$

Therefore from equations (19) and, (21) (16) becomes

$$R \leq \lim_{n\to\infty}\frac{1}{n}H\big(\mathcal{Y}'\big). \quad (22)$$

As before, let K' be the number of elements in $\mathcal{Y}'$ and let the random variable $N_1, N_2, \ldots, N_{K'}$ be the lengths of the islands. The result in [29] gives us a way to upper bound this entropy term. It showed that the entropy of unordered sets of variable length binary strings can be upper bounded by the difference of two terms as "cumulative coverage depth − reordering cost", two terms that were introduced in [29]. Precisely it showed that for an unordered set $\mathcal{Y}'$ with binary strings of lengths given by $N_1, N_2, \ldots, N_{K'}$, where K' is also a random variable, we have

$$\lim_{n\to\infty}\frac{1}{n}H\big(\mathcal{Y}'\big) \leq \lim_{n\to\infty}\left(\frac{1}{n}E\big[K'\big]E[N_1] - E\big[K'\big]\frac{\log n}{n}\right)$$
$$\overset{(a)}{=} \big(1-e^{-c}\big) - \frac{c}{\bar{L}}e^{-c},$$

when $\lim_{n\to\infty}\frac{\log n}{E[N_i]} \in (0,\infty)$ and $E[N_1^2/(\log n)^2]$ is finite and bounded. This is indeed true and is proved in Appendix G. Here (a) is due to Lemmas 1 and 2.

This is an upper bound to the achievable rate we seek to match from Section IV. Figure 3 shows that this is in fact a strict upper bound. This intuitively indicates that the genie

we use is probably too powerful. This makes sense, since discerning smaller overlaps probably adds a cost. In fact, our achievable rate calculated this cost (see Lemma 5).

Constrained genie: Let's re-introduce a genie, except this genie can only merge reads with overlap sizes of at least $\delta \log n$. Following the nomenclature in [2], we call these *apparent islands*. Let $\mathcal{Y}'_\delta$ be the set formed by this genie. Following the same steps as above, we can say that

$$R \leq \lim_{n\to\infty} \frac{1}{n} H\big(\mathcal{Y}'_\delta\big). \tag{23}$$

Let K'' be the number of apparent islands formed. Let $N_1^\delta, \ldots, N_{K''}^\delta$ be the length of the islands. Like before we use the result in [12] to bound (23). We omit the many steps that are exactly the same as that for the omniscient genie. Employing the result in [29] again, we obtain

$$\lim_{n\to\infty} \frac{1}{n} H\big(\mathcal{Y}'_\delta\big) \leq \lim_{n\to\infty} \left(\frac{1}{n} E\big[K''\big] E[N_1] - E\big[K''\big] \frac{\log n}{n} \right). \tag{24}$$

Setting $\sigma := 1 - \frac{\delta}{\bar{L}}$, it can be shown that

$$\begin{aligned} &\lim_{n\to\infty} \left(\frac{1}{n} E[K''] E[N_1] - E\big[K''\big] \frac{\log n}{n} \right) \\ &\quad = \big(1 - e^{-c\sigma}\big) + c(1-\sigma) e^{-c\sigma} - \frac{c}{\bar{L}} e^{-c\sigma}. \end{aligned} \tag{25}$$

The proof of this is presented in Appendix H. Therefore,

$$\lim_{n\to\infty} \frac{1}{n} H\big(\mathcal{Y}'_\delta\big) \leq \big(1 - e^{-c\sigma}\big) + c(1-\sigma) e^{-c\sigma} - \frac{c}{\bar{L}} e^{-c\sigma}. \tag{26}$$

The final step is to minimize this bound over σ. The minimum value of (26) is achieved when $\sigma = 1 - \frac{1}{\bar{L}}$ (or $\delta = 1$). Therefore, substituting this value, we obtain

$$\lim_{n\to\infty} \frac{1}{n} H\big(\mathcal{Y}'\big) \leq 1 - e^{-c\left(1 - \frac{1}{\bar{L}}\right)}. \tag{27}$$

An interesting observation here is that $\delta = 1$ implies the constrained genie should merge all reads with overlap sizes greater than $\log n$ for the converse to match the achievable rate. This is consistent with Lemma 3, which implies that multiple merge candidates are likely to exist when the overlap size is less than or equal to $\log n$. This completes the converse proof.

VI. Conclusion

In this work, motivated by applications in DNA data storage, we introduced the Shotgun Sequencing Channel (SSC). We characterized the SSC capacity exactly. This capacity was shown to strictly upper bound the capacity of a shuffling-sampling [12] channel, which modelled DNA storage systems that sampled uniformly at random from a set of short length strings. In fact, we showed high gains for the same, by showing that the capacity of the SSC goes to one for high coverage depth, allowing highly reliable reconstruction of the string, even for short read lengths $L \approx \log n$. In contrast, for the shuffling-sampling channel [12] the capacity goes to zero in the same regime, implying we would not be able to recover the set of short strings in the same regime. This reveals the high capacity gains that would be achieved if data could be stored in a long DNA molecule instead of short-length molecules.

We further showed that when we shotgun sequence a DNA string that is a codeword from a codebook, we can reduce the minimum required read length by a factor of 2 and the number of reads sampled by a factor of $\log n$, while still admitting arbitrarily close to perfect reconstruction.

DNA synthesis and storage in general admit errors that can cause bit flips and erasures. This has been studied before in the context of the shuffling-sampling channel [13], [16], [18]. This motivates studying a noisy version of the SSC, where the noise can be modeled as concatenated binary symmetric or erasure channels. We also note that our analysis is restricted to the asymptotic regime $n \to \infty$. It may be of interest to see the reliability of such systems in the finite blocklength regime.

Appendix A
Proof of Lemma 1

To prove the above result we first compute $E[\Phi]$ as follows: (We define $A_i := \{X_i \text{ is covered by } \mathcal{Y}\}$)

$$\begin{aligned} E[\Phi] &= \frac{1}{n} \sum_{i=1}^{n} E\big[\mathbf{1}_{A_i}\big] = \Pr(A_n) = 1 - \Pr\big(A_n^c\big) \\ &= 1 - \Pr\big(X_n \text{ is not covered by } \vec{Y}_i,\ \forall i \in [1:K]\big) \\ &\overset{(a)}{=} 1 - \Pr\big(X_n \text{ is not covered by } \vec{Y}_1\big)^K \\ &\overset{(b)}{=} 1 - (1 - L/n)^K \to 1 - e^{-c}, \end{aligned} \tag{28}$$

as $n \to \infty$. Here, (a) is due the starting points being picked independently and uniformly at random, and (b) is due to the fact that X_n needs to start in L (contiguous) bits out of n total bits to cover X_n. From the definition of limit, we know that there for every $\epsilon > 0$, there exists an N such that, for $n \geq N$,

$$\big|E[\Phi] - \big(1 - e^{-c}\big)\big| < \epsilon\big(1 - e^{-c}\big)/2.$$

If it also holds that $|\Phi - E[\Phi]| < \epsilon(1 - e^{-c})/2$, then by triangle's inequality, we obtain $|\Phi - (1 - e^{-c})| \leq |\Phi - E[\Phi]| + |E[\Phi] - (1 - e^{-c})| \leq \epsilon(1 - e^{-c})$, for sufficiently large n. Defining $\epsilon' = \epsilon(1 - e^{-c})/2$ and using Chebyshev's inequality for n large enough, we have

$$\begin{aligned} &\Pr\big(\big|\Phi - \big(1 - e^{-c}\big)\big| > \epsilon\big(1 - e^{-c}\big)\big) \\ &\quad \leq \Pr\big(|\Phi - E[\Phi]| > \epsilon'\big) \leq \frac{\mathrm{Var}(\Phi)}{\epsilon'^2}. \end{aligned} \tag{29}$$

Let's calculate $\mathrm{Var}(\Phi)$. By linearity of covariance, we have

$$\mathrm{Var}(\Phi) = \frac{1}{n^2} \sum_{i,j} \mathrm{Cov}\big(\mathbf{1}_{A_i} \mathbf{1}_{A_j}\big). \tag{30}$$

Now, note that for $(i - j) \mod n > \bar{L} \log n$, we can calculate the covariance as

$$\begin{aligned} &\mathrm{Cov}\big(\mathbf{1}_{A_i} \mathbf{1}_{A_j}\big) = E\big[\mathbf{1}_{A_i} \mathbf{1}_{A_j}\big] - E\big[\mathbf{1}_{A_i}\big] E\big[\mathbf{1}_{A_j}\big] \\ &\overset{(a)}{\leq} \left(1 - \left(1 - \frac{L}{n}\right)^K\right)\left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right) - \left(1 - \left(1 - \frac{L}{n}\right)^K\right)^2 \\ &\leq 0. \end{aligned}$$

Here, (a) is because there exists at least one read that has to cover X_i and not X_j when $(i - j) \mod n > \bar{L} \log n$. Therefore,

we can upper bound (30) as

$$\begin{aligned}\mathrm{Var}(\Phi) &\le \frac{1}{n^2}\left(\sum_{i=1}^{n}\mathrm{Var}\big(\mathbf{1}_{A_i}\big) + \sum_{i,j\,:\,((i-j) \bmod n \le \bar{L}\log n)}\mathrm{Cov}\big(\mathbf{1}_{A_i}\mathbf{1}_{A_j}\big)\right)\\ &\le \frac{1}{n^2}\big(n+\bar{L}n\log n\big).\end{aligned}$$

This leads to (29) being upper bounded as

$$\Pr\big(|\Phi - E[\Phi]| > \epsilon'\big) \le \frac{\mathrm{Var}(\Phi)}{\epsilon'^2} \le \left(\frac{1}{n}+\frac{\bar{L}\log n}{n}\right)\frac{1}{\epsilon'^2} \to 0, \tag{31}$$

as $n \to \infty$, which completes the proof.

Appendix B
Proof of Lemma 2

To prove the above lemma, first we note that we can count the number of islands by counting the reads that have no overlap, since any read with no (suffix) overlap must be the last read of an island. Therefore, we have (We define $P_i := \{\vec{Y}_i$ has no overlap$\}$)

$$E\big[K'\big] = \sum_{i=1}^{K} E\big[\mathbf{1}_{P_i}\big] = K\Pr(P_1) = K\left(1-\frac{L}{n}\right)^{K-1}. \tag{32}$$

Note that $\lim_{n\to\infty}\frac{\log n}{n}E[K'] = \frac{c}{L}e^{-c}$. Thus, we can say (from the definition of limit) that for any $\epsilon/2 > 0$, $|E[K'] - Ke^{-c}| < Ke^{-c}\epsilon/2$, for n large enough. Therefore if $|K' - E[K']| < Ke^{-c}\epsilon/2$, then by triangle's inequality$|K' - Ke^{-c}| \le |K' - E[K']| + |E[K'] - Ke^{-c}| \le Ke^{-c}\epsilon$, for n large enough. Using Chebyshev's inequality, for n large enough we have that

$$\begin{aligned}&\Pr\big(\big|K' - Ke^{-c}\big| \ge \epsilon Ke^{-c}\big)\\ &\quad\le \Pr\big(\big|K' - E\big[K'\big]\big| \ge \epsilon' Ke^{-c}\big) \le \frac{\mathrm{Var}\big(K'\big)}{\epsilon'^2K^2e^{-2c}} \to 0,\end{aligned} \tag{33}$$

as $n \to \infty$ and $\epsilon' = \epsilon/2$. To see this, we can calculate $\mathrm{Var}(K')$ as $\mathrm{Var}(K') = \sum_{i,j}\mathrm{Cov}(\mathbf{1}_{P_i}\mathbf{1}_{P_j})$. If $i \ne j$, then we can compute the covariances as follows (we define the event $Q_{i,j} := \{\vec{Y}_i$ and $\vec{Y}_j$ overlap with each other$\}$):

$$\begin{aligned}\mathrm{Cov}\big(\mathbf{1}_{P_i}\mathbf{1}_{P_j}\big) &= E\big[\mathbf{1}_{P_i}\mathbf{1}_{P_j}\big] - E\big[\mathbf{1}_{P_1}\big]^2 \overset{(a)}{\le} \Pr(Q_{i,j})\\ &\quad+ \Pr\Big(\vec{Y}_i, \vec{Y}_j \text{ don't have overlaps}|Q_{i,j}^c\Big) - \left(1-\frac{L}{n}\right)^{2(K-1)}\\ &\overset{(b)}{\le} \left(1-\frac{L}{n}\right)^{2(K-2)} + \frac{k\log n}{n} - \left(1-\frac{L}{n}\right)^{2(K-1)} \le 0,\end{aligned}$$

for n large enough and a fixed constant k. (a) is due the Law of total probability and (b) is because conditioned on the fact that two reads don't overlap with each other, the probability that they don't overlap are independent of one another. We calculate the worst case probability of the rest of the reads overlapping with them. This can be done assuming there $K-2$ reads left at most for each. Therefore for n large enough, we can say that $\mathrm{Var}(K') \le \sum_{i=1}^{K}\mathrm{Var}(\mathbf{1}_{P_i}) \le K$. Therefore, (33) can be upper bounded as

$$\Pr\big(\big|K' - Ke^{-c}\big| \ge \epsilon Ke^{-c}\big) \le \frac{1}{\epsilon'^2 Ke^{-2c}} \to 0, \tag{34}$$

as $n \to \infty$.

Appendix C
Proof of Lemma 3

We will first prove the first part of the lemma and then the second part of the lemma. Both proofs are similar with some small variations. We first look at a concentration result on the number of times $\vec{z}$ appears on a length-n i.i.d. Bern$(1/2)$ string X^n. We will then use this to prove Lemma 3. Let $N_{\vec{z}} = \sum_{i=1}^{n}\mathbf{1}_{\{\vec{z} \text{ present starting at the } i\text{th symbol of } X^n\}} := \sum_{i=1}^{n}\mathbf{1}_{\mathrm{F}_i}$. Note that $E[N_{\vec{z}}] = n \times 2^{-\gamma(\vec{z})\log n} = n^{1-\gamma(\vec{z})}$.

Lemma 7: For all $\epsilon > 0$,

$$\Pr\left(\left|N_{\vec{z}} - n^{1-\gamma(\vec{z})}\right| \ge \epsilon n^{1-\gamma(\vec{z})}\right) \le \big(2\bar{L}\log n\big)e^{-(n^{\epsilon})\epsilon^2/(2\bar{L}\log n)}. \tag{35}$$

Proof: We can rewrite $N_{\vec{z}}$ as

$$\begin{aligned}N_{\vec{z}} = \sum_{i=1}^{n}\mathbf{1}_{F_i} &= \underbrace{\sum_{t=0}^{\frac{n}{\gamma(\vec{z})\log n}-1}\mathbf{1}_{F_{1+t\gamma(\vec{z})\log n}}}_{N_{1,\vec{z}}} + \underbrace{\sum_{t=0}^{\frac{n}{\gamma(\vec{z})\log n}-1}\mathbf{1}_{F_{2+t\gamma(\vec{z})\log n}}}_{N_{2,\vec{z}}}\\ &\quad+\cdots+\underbrace{\sum_{t=0}^{\frac{n}{\gamma(\vec{z})\log n}-1}\mathbf{1}_{F_{\gamma(\vec{z})\log n+t\gamma(\vec{z})\log n}}}_{N_{\gamma(\vec{z})\log n,\vec{z}}}.\end{aligned}$$

Notice that each summation deals with starting locations that are at least $\gamma(\vec{z})\log n$ symbols apart, and the resulting random variables $\mathbf{1}_{F_i}$ are independent. Now note that by triangle inequality if each of

$$R_i := \left|N_{i,\vec{z}} - \frac{n^{1-\gamma(\vec{z})}}{\gamma(\vec{z})\log n}\right| \le \epsilon\frac{n^{1-\gamma(\vec{z})}}{\gamma(\vec{z})\log n} \tag{36}$$

then $|N_{\vec{z}} - n^{1-\gamma(\vec{z})}| \le \epsilon n^{1-\gamma(\vec{z})}$. Employing the union bound, we can say that

$$\begin{aligned}&\Pr\left(\left|N_{\vec{z}} - n^{1-\gamma(\vec{z})}\right| \ge \epsilon n^{1-\gamma(\vec{z})}\right) \le \sum_{i=1}^{\gamma(\vec{z})\log n}\Pr\big(R_i^c\big)\\ &\le \big(\bar{L}\log n\big)\Pr\big(R_1^c\big) \overset{(a)}{\le} \big(2\bar{L}\log n\big)e^{\frac{-n\times n^{-\gamma(\vec{z})}\epsilon^2}{2\gamma(\vec{z})(\log n)\left(1-n^{-\gamma(\vec{z})}\right)}}\\ &\le \big(2\bar{L}\log n\big)e^{-n^{1-\gamma(\vec{z})}\epsilon^2/(2\bar{L}\log n)} \overset{(b)}{\le} \big(2\bar{L}\log n\big)e^{-n^{\epsilon}\epsilon^2/(2\bar{L}\log n)}\end{aligned} \tag{37}$$

where step (b) holds for all $\gamma(\vec{z}) \le 1-\epsilon$. Here (a) is due to the following form of Hoeffding's inequality:

Lemma 8: For i.i.d. Bernoulli $X_1, X_2 \ldots X_n$ with parameter p,

$$\Pr\left(\left|\frac{1}{n}\sum_{i=1}^{n}X_i - p\right| \ge \epsilon p\right) \le 2e^{-nD(p+\epsilon||p)} \le 2e^{-np\epsilon^2/(1-p)} \tag{38}$$

This ends the proof of Lemma 7. ■

Now given the string X^n, define $\mathcal{D} :=$ {Starting point indices of $\vec{z}$ in X^n}.

Therefore $M_{\vec{z}} = \sum_{i=1}^{K}\mathbf{1}_{\{i\text{th read has starting point in } \mathcal{D}\}}$. Since the generation of these reads is independent and uniformly at

random, the random variables in the summation are independent of each other (given $\mathcal{D}$). Now we can say (We define $S = \{|N_{\vec{z}} - n^{1-\gamma(\vec{z})}| \le \epsilon n^{1-\gamma(\vec{z})}\}$)

$$\begin{aligned}
&\Pr\left(\bigcup_{z\in\mathcal{Z},\gamma(\vec{z})<1}\left\{\left|M_{\vec{z}} - Kn^{-\gamma(\vec{z})}\right| \ge \epsilon Kn^{-\gamma(\vec{z})}\right\}\right)\\
&\overset{(a)}{\le} dn \max_{\vec{z}\in\mathcal{Z},\gamma(\vec{z})<1}\Pr\left(\left\{\left|M_{\vec{z}} - Kn^{-\gamma(\vec{z})}\right| \ge \epsilon Kn^{-\gamma(\vec{z})}\right\}\right)\\
&\overset{(b)}{\le} dn \max_{\vec{z}\in\mathcal{Z},\gamma(\vec{z})<1}\left(\Pr\left(\left\{\left|M_{\vec{z}} - Kn^{-\gamma(\vec{z})}\right| \ge \epsilon Kn^{-\gamma(\vec{z})}\right\}\middle|S\right)\right)\\
&\quad + \Pr(S^c)) \overset{(c)}{\le} dn\left(2e^{-\frac{cn^{\epsilon}}{2\bar{L}\log n}\epsilon^2} + (2\bar{L}\log n)e^{-n^{\epsilon}\epsilon^2/2\bar{L}\log n}\right)\\
&\to 0, \qquad (39)
\end{aligned}$$

as $n \to \infty$. Here (a) is due to the union bound on the number of $\vec{z}$ for $\gamma(\vec{z}) \le 1-\epsilon$ (which is $\le dn$, for some fixed constant d, since there are at most $\sum_{\gamma\le 1-\epsilon} 2^{\gamma\log n} = \sum_{\gamma\le 1-\epsilon} n^{\gamma}$ vectors of that size, and for $\gamma \le 1-\epsilon$, $\sum_{\gamma\le 1-\epsilon} n^{\gamma} \le n^{1-\epsilon}\log n \le dn$, for n large enough), (b) is from the law of total probability and (c) is due to the Hoeffding's inequality. To see the last step note that $\mathbf{1}_{\{i\text{th read starts in }\mathcal{D}\}}$ is a Bern(p) RV. When conditioned on the event $|N_{\vec{z}} - n^{1-\gamma(\vec{z})}| \le \epsilon n^{1-\gamma(\vec{z})}$, $p \in [(1-\epsilon)n^{-\gamma(\vec{z})}, (1+\epsilon)n^{-\gamma(\vec{z})}]$. We can apply Hoeffding's inequality assuming the worst case of $p = (1+\epsilon)n^{-\gamma(\vec{z})}$. Now since $\gamma \le 1-\epsilon$ in the above case, this implies that $np \le n\times(1+\epsilon)n^{-\gamma(\vec{z})} \le (1+\epsilon)n^{\epsilon}$. Now apply this bound on np to Lemma 8, to obtain the result.

Now for the second part of the proof we employ a similar trick as before. Using the same split on $N_{\vec{z}}$ as before we have that

$$\begin{aligned}
&\Pr\left(N_{\vec{z}} \ge \frac{L\times n^{\epsilon}}{c}\right) \overset{(a)}{\le} (\bar{L}\log n)\Pr\left(N_{1,\vec{z}} \ge \frac{n^{\epsilon}}{c}\right)\\
&= \left(\bar{L}\log n\right)\Pr\left(N_{1,\vec{z}} - \frac{n^{1-\gamma}}{L} \ge \frac{n^{\epsilon}}{c} - \frac{n^{1-\gamma}}{L}\right)\\
&\le \left(\bar{L}\log n\right)\exp\left(-\frac{\left(n^{\epsilon}/c - n^{1-\gamma}/L\right)^2}{2n^{1-\gamma}\left(1-n^{-\gamma}\right)}\right)\\
&\le \left(\bar{L}\log n\right)\exp\left(-n^{(2\epsilon-(1-\gamma'))}\frac{\left(1/c - n^{1-\gamma-\epsilon}/L\right)^2}{2}\right), \qquad (40)
\end{aligned}$$

where (a) is due to the union bound, specifically because $\gamma \le \bar{L}$, which implies that there are at most $\bar{L}\log n$ terms. Here we use the following version of Hoeffding's inequality:

Lemma 9: If $X_1, \ldots, X_n$ are i.i.d. Ber(p) random variables,

$$\Pr\left(\sum_{i=1}^{n} X_i - np \ge x\right) \le e^{-x^2/(2np(1-p))} \qquad (41)$$

We now use (40), to condition the event $\{M_{\vec{z}} \ge n^{\epsilon}\}$ on $N_{\vec{z}} \le (Ln^{\epsilon})/c$ and follow an analysis similar to (39), to arrive at

$$\Pr\left(\bigcup_{z\in\mathcal{Z}\,:\,\gamma(\vec{z})>1-\epsilon}\left\{M_{\vec{z}} \ge n^{\epsilon}\right\}\right) \to 0,$$

as $n \to \infty$.

Appendix D
Proof of Lemma 4

Proof: We use Chebyshev's inequality to see that

$$\Pr\left(|G(\gamma) - \bar{G}(\gamma)| \ge \epsilon\bar{G}(\gamma)\right) \le \frac{\mathrm{Var}(G(\gamma))}{\epsilon^2\bar{G}(\gamma)^2}. \qquad (42)$$

This can be rewritten using the fact that

$$\frac{\mathrm{Var}(G(\gamma))}{\bar{G}(\gamma)^2} = \frac{E\left[G(\gamma)^2\right]}{\bar{G}(\gamma)^2} - 1. \qquad (43)$$

We also have that (Define $J_{i,j} \coloneqq E[G(\gamma)_i G(\gamma)_j]$)

$$\begin{aligned}
E\left[G(\gamma)^2\right] &= E\left[\left(\sum_{i=1}^{K} G(\gamma)_i\right)^2\right] = \left(\sum_{i=1}^{K} E\left[G(\gamma)_i^2\right]\right)\\
&\quad + \sum_{i\ne j} J_{i,j} \le KJ_{1,1} + K^2J_{1,2} = KE\left[G(\gamma)_1\right] + K^2J_{i,j}\\
&= \bar{G}(\gamma) + K^2E\left[G(\gamma)_1G(\gamma)_2\right]. \qquad (44)
\end{aligned}$$

Recalling that the definition of "overlap", implies an overlap only between successive reads (Definition 1) and focusing on the second term in the above equation, we have (Recall $Q_{i,j} \coloneqq \{\vec{Y}_i$ and $\vec{Y}_j$ overlap with each other$\}$)

$$\begin{aligned}
&E\left[G(\gamma)_1G(\gamma)_2\right] = \Pr(Y_1, Y_2 \text{ both have overlap size } \gamma\log n)\\
&\le \Pr\left(Y_1, Y_2 \text{ both have overlap size } \gamma\log n | Q_{1,2}^c\right)\\
&\quad + \Pr(Q_{1,2})\\
&\overset{(a)}{\le} \Pr\left(Y_1, Y_2 \text{ both have overlap size } \gamma\log n | Q_{1,2}^c\right) + \frac{k\log n}{n}\\
&\overset{(b)}{\le} \frac{\bar{G}(\gamma)^2}{K^2} + k\frac{\log n}{n}, \qquad (45)
\end{aligned}$$

where k is a finite constant. Here (a) is because two reads with lengths $\bar{L}\log n$ have an overlap if their starting points T_1, T_2 are such that $T_2 - T_1 \le L$ and (b) is because the conditioning constrains the starting points to not be such that $T_2 - T_1 \le L$, implying that the events $\{Y_i$ has an overlap of size $\gamma\log n\}$ for $i = 1, 2$ are conditionally independent and with probability at most $E[G(\gamma)_1] = \bar{G}(\gamma)/K$. Therefore we have

$$E\left[G(\gamma)^2\right] \le \bar{G}(\gamma) + \bar{G}(\gamma)^2 + K^2k\frac{\log n}{n}. \qquad (46)$$

Hence,

$$\begin{aligned}
\frac{\mathrm{Var}(G(\gamma))}{\bar{G}(\gamma)^2} &= \frac{E\left[G(\gamma)^2\right]}{\bar{G}(\gamma)^2} - 1 \le \frac{1}{\bar{G}(\gamma)} + 1 - 1 + \frac{K^2k\log n}{n\bar{G}(\gamma)^2}\\
&= \frac{1}{\bar{G}(\gamma)} + \frac{K^2k\log n}{n\Theta\left(n^2/\log^4 n\right)}\\
&\overset{(a)}{\le} \frac{n}{(K-1)^2\left(1 - \frac{(\bar{L}-\gamma)\log n+1}{n}\right)^{(K-2)}} + \Theta\left(n^{-1}\log^3 n\right)\\
&\le \frac{n}{(K-1)^2\left(1 - \frac{\bar{L}\log n+1}{n}\right)^{(K-2)}} + \Theta\left(n^{-1}\log^3 n\right)\\
&= \Theta\left(\frac{\log^3 n}{n}\right), \qquad (47)
\end{aligned}$$

where (a) is due to the lower bound in (50), followed by further noting that $K-1 \le K$. Now due to the union bound

$$\Pr\left(\bigcup_{\gamma\in\Gamma}\left\{|G(\gamma)-\bar{G}(\gamma)| \ge \epsilon\bar{G}(\gamma)\right\}\right) \le \frac{\bar{L}\log n}{\epsilon^2}\cdot\Theta\left(\frac{\log^3 n}{n}\right).$$

The above expression $\to 0$ as $n\to\infty$. ∎

Appendix E
Proof of Lemma 5

From the definition of $\overline{CI}_n$,

$$\begin{aligned}\overline{CI}_n &= P\times\prod_{\gamma\le 1-\epsilon}B_3(\gamma)^{B_4(\gamma)}\times\prod_{\gamma>1-\epsilon}n^{\epsilon B_4(\gamma)} = (L+1)^K\\ &\quad\times(1+\epsilon)^{(1-\epsilon)\bar{L}\log n}n^{(1+\epsilon)\sum_{\gamma\le 1-\epsilon}(1-\gamma)\bar{G}(\gamma)}\\ &\quad\times n^{\epsilon(1+\epsilon)\sum_{\gamma>1-\epsilon}\bar{G}(\gamma)}. \end{aligned}\tag{48}$$

Hence, we have that (We define $g(\epsilon)\coloneqq\epsilon(1+\epsilon)$)

$$\begin{aligned}&\lim_{n\to\infty}\frac{1}{n}\log\overline{CI}_n\\ &= \lim_{n\to\infty}\left(\frac{K\log(L+1)}{n}+\frac{\bar{L}\log n}{n}(1-\epsilon)\log(1+\epsilon)\right.\\ &\qquad+\frac{\log n}{n}(1+\epsilon)\sum_{\gamma\le 1-\epsilon}(1-\gamma)\bar{G}(\gamma)\\ &\qquad\left.+\frac{\epsilon(1+\epsilon)\log n}{n}\sum_{\gamma>1-\epsilon}\bar{G}(\gamma)\right)\\ &= \lim_{n\to\infty}\frac{\log n}{n}(1+\epsilon)\sum_{\gamma\le 1-\epsilon}(1-\gamma)\bar{G}(\gamma)\\ &\quad+g(\epsilon)\lim_{n\to\infty}\frac{\log n}{n}\sum_{\gamma>1-\epsilon}\bar{G}(\gamma).\end{aligned}\tag{49}$$

Notice that second term in (49) satisfies

$$\begin{aligned}0 &\le \epsilon(1+\epsilon)\lim_{n\to\infty}\frac{\log n}{n}\sum_{\gamma>1-\epsilon}\bar{G}(\gamma) \le \epsilon(1+\epsilon)\lim_{n\to\infty}\frac{\log n}{n}K\\ &= \epsilon(1+\epsilon)\frac{c}{\bar{L}}\to 0,\end{aligned}$$

as $\epsilon\to 0$.

Now we look at the first term in (49). Before we proceed to evaluate the required summation, we need to calculate $\bar{G}(\gamma)$. Calculating this exactly is difficult, but we can find upper and lower bounds that asymptotically converge. We start by noticing that (We define the function $R(X_1,X_2\ldots X_n)$ be the min start location of $(X_1,X_2\ldots X_n)$)

$$\begin{aligned}\bar{G}(\gamma) &= K\Pr(X_1 \text{ has an overlap of size } \gamma\log n)\\ &= K\Pr(X_1 \text{ has an overlap of size } \gamma\log n | X_1 \text{ starts from 1})\\ &= K\Pr\big(R(X_2,\ldots,X_K)=\big(\bar{L}-\gamma\big)\log n | X_1 \text{ starts from 1}\big).\end{aligned}$$

Let's look at Pr(min start location of $(X_2,\ldots,X_K) = (\bar{L}-\gamma)\log n | X_1$ starts from 1). We can upper and lower bound this probability by forcing one read to start at position $(\bar{L}-\gamma)\log n$ and all others to start at position $(\bar{L}-\gamma)\log n$ or higher (which will lead to double counting, and thus an upper bound). For the lower bound we can assume that exactly one of the reads starts at $(\bar{L}-\gamma)\log n$ and the rest start at positions strictly greater than $(\bar{L}-\gamma)\log n$. Thus we get

$$\begin{aligned}K(K-1)\times\frac{1}{n}\left(1-\frac{(\bar{L}-\gamma)\log n+1}{n}\right)^{K-2} &\le \bar{G}(\gamma)\\ \le K(K-1)\times\frac{1}{n}\left(1-\frac{(\bar{L}-\gamma)\log n}{n}\right)^{K-2}.\end{aligned}\tag{50}$$

Pick $\Delta_n = \gamma+\frac{1}{\log n}-\gamma$. Now we can say that

$$\begin{aligned}\lim_{n\to\infty}\frac{\log n}{n}\sum_{\gamma\le 1-\epsilon}(1-\gamma)\bar{G}(\gamma) &= \lim_{n\to\infty}\sum_{\gamma\le 1-\epsilon}(1-\gamma)\frac{\bar{G}(\gamma)\log n}{\Delta_n n}\Delta_n\\ &\overset{(a)}{=}\int_0^{1-\epsilon}(1-\gamma)\lim_{n\to\infty}\left(\bar{G}(\gamma)\frac{(\log n)^2}{n}\right)d\gamma,\end{aligned}\tag{51}$$

where (a) follows from the definition of Riemann integration. Let us evaluate the limit inside the integral first. Applying the Sandwich theorem to (50), we have that

$$\lim_{n\to\infty}\bar{G}(\gamma)\frac{(\log n)^2}{n} = \frac{c^2}{\bar{L}^2}\exp\left(-c\left(1-\frac{\gamma}{\bar{L}}\right)\right).$$

Therefore from (51) we have that

$$\begin{aligned}&\lim_{n\to\infty}\frac{\log n}{n}\sum_{\gamma\le 1-\epsilon}(1-\gamma)\bar{G}(\gamma)\\ &= \int_0^{1-\epsilon}(1-\gamma)\lim_{n\to\infty}\left(\bar{G}(\gamma)\frac{(\log n)^2}{n}\right)d\gamma\\ &= \frac{c^2}{\bar{L}^2}\int_0^{1-\epsilon}(1-\gamma)\exp\left(-c\left(1-\frac{\gamma}{\bar{L}}\right)\right)d\gamma\\ &\le \frac{c^2}{\bar{L}^2}\int_0^{1}(1-\gamma)\exp\left(-c\left(1-\frac{\gamma}{\bar{L}}\right)\right)d\gamma\\ &= \frac{c^2}{\bar{L}^2}e^{-c\left(1-\frac{1}{\bar{L}}\right)}\int_0^1 ze^{-\frac{c}{\bar{L}}z}dz\\ &= e^{-c\left(1-\frac{1}{\bar{L}}\right)}\left(1-\left(\frac{c}{\bar{L}}+1\right)e^{-\left(\frac{c}{\bar{L}}\right)}\right) = e^{-c\left(1-\frac{1}{\bar{L}}\right)}-\left(\frac{c}{\bar{L}}+1\right)e^{-c},\end{aligned}$$

where we used the substitution $z=1-\gamma$. Finally, plugging everything back into (49), we obtain

$$\begin{aligned}\lim_{n\to\infty}\frac{1}{n}\log\overline{CI}_n &\le (1+\epsilon)\left(e^{-c\left(1-\frac{1}{\bar{L}}\right)}-\left(\frac{c}{\bar{L}}+1\right)e^{-c}\right)+g(\epsilon)\frac{c}{\bar{L}}\\ &= e^{-c\left(1-\frac{1}{\bar{L}}\right)}-\left(\frac{c}{\bar{L}}+1\right)e^{-c}+f(\epsilon),\end{aligned}$$

where $f(\epsilon)\to 0$ as $\epsilon\to 0$, concluding the proof.

Appendix F
Proof of Lemma 6

We first upper bound the $\Pr(D>\gamma_0\log n)$ as

$$\begin{aligned}&\Pr(D>\gamma_0\log n)\\ &\quad\le K\Pr(\text{No. of samples in a given island } > \gamma_0\log n)\end{aligned}\tag{52}$$

and then show that this upper bound tends to zero as $n\to\infty$. We first define a few terms. Let the sequence $\vec{Y}_1,\vec{Y}_2,\ldots,$ represent the reads (in an ordered fashion) in the given island. We define $U_i \coloneqq T_{i+1}-T_i$ with T^K being the vector of ordered starting locations, as the separation between read $\vec{Y}_i$

and $\vec{Y}_{i+1}$. We can thus think of the sampling scheme as a random process that picks starting locations T^K with inter-arrival times $U_1, U_2, \ldots, U_{K-1}$. For convenience, we define $U_K = ((T_1 - T_K) \mod n) + 1$ (this is to capture the wrap around nature of the reads). Note that $\sum_{i=1}^{K} U_i = n$.

Thus we calculate Pr(A given read has an overlap) as follows,

$$\begin{aligned}&\Pr(\text{A given read has an overlap})\\ &= 1 - \Pr(\text{A given read has no overlap}) = 1 - \left(1 - \frac{L}{n}\right)^{K-1}.\end{aligned} \tag{53}$$

We now use the above quantity to calculate (52). We note that

$$\begin{aligned}&\Pr(\text{No. of samples in a given island } > \gamma_0 \log n)\\ &= \Pr\big(\text{No. of samples in a given island}\\ &\qquad > \gamma_0 \log n \big| \text{given island starts from } \vec{Y}_1\big)\\ &= \Pr\big(U_1 \le L, U_2 \le L, \ldots, U_{\gamma_0 \log n - 1} \le L\big)\\ &= \Pr(U_1 \le L) \times \Pr(U_2 \le L | U_1 \le L)\\ &\quad \times \cdots \times \Pr\big(U_{\gamma_0 \log n - 1} | U_1 \le L, U_2 \le L \times \ldots U_{\gamma_0 \log n - 2} \le L\big)\\ &\overset{(a)}{\le} \Pr(U_1 \le L) \Pr(U_2 \le L) \ldots \Pr\big(U_{\gamma_0 \log n - 1} \le L\big)\\ &= \left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right)^{\gamma_0 \log n - 1},\end{aligned} \tag{54}$$

where (a) is because the $U_i, i \in [1:K]$ are negatively associated (because the larger one U_i is, the less room there is on x^n for the other U_j's). For negatively associated random variables we know that [32]

$$\Pr(U_i \le L : i \in [1:K]) \le \Pi_{i=1}^{K} \Pr(U_i \le L) \tag{55}$$

Therefore (52) can be upper bounded as

$$\begin{aligned}\Pr(D \ge \gamma_0 \log n) &\le K\left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right)^{\gamma_0 \log n - 1}\\ &= \frac{c \times 2^{\log n + (\gamma_0 \log n - 1) \log\left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right)}}{\bar{L} \log n}\\ &= \frac{c \times 2^{\log n\left(1 + \gamma_0 \log\left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right)\right) - \log\left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right)}}{\bar{L} \log n}.\end{aligned} \tag{56}$$

Now as long as

$$\lim_{n\to\infty}\left(1 + \gamma_0 \log\left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right)\right) < 0 \text{ or}$$

$$\gamma_0 > \frac{-1}{\log\left(1 - e^{-c}\right)},$$

Equation (56) $\to 0$ as $n \to \infty$.

Appendix G
Finiteness of Island Lengths

We are required to prove that (a) $\lim_{n\to\infty} \frac{\log n}{E[N_1]} \in (0, \infty)$ and (b) $E[N_1^2/(\log n)^2]$ is finite and bounded. We note that $N_i = \sum_{i=1}^{J} Z_i$, where J is the random variable which indicates the number of reads in an island and Z_i is the length of the reads after removing the overlapping part of the read. To see (a) note that there are K' islands. Define $J_1, J_2, \ldots, J_{K'}$ as the number of reads in each of these islands. Note that J and J_i are identically distributed for all i. Therefore,

$$K = \sum_{i=1}^{K'} J_i.$$

Note that K' is a stopping time with respect to $J_1, J_2, \ldots$. This implies that $E[K']E[J] = K$. Also note that J is a stopping time with respect to $Z_1, Z_2 \ldots$ Therefore we can say

$$\begin{aligned}\lim_{n\to\infty} \frac{\log n}{E[N_1]} &= \lim_{n\to\infty} \frac{\log n}{E[J]E[Z_1]} = \lim_{n\to\infty} \frac{E\big[K'\big] \log n}{KE[Z_1]}\\ &= \left(\lim_{n\to\infty} \frac{E[K']}{K}\right)\left(\lim_{n\to\infty} \frac{\log n}{E[Z_1]}\right) \overset{(a)}{=} \frac{ce^{-c}}{\bar{L}} d \in (0, \infty),\end{aligned}$$

where d is a fixed constant, since $E[Z_1] \sim \Theta(\log n))$. To handle (b) we note the following

$$E\Big[N_i^2\Big] = E\left[\left(\sum_{i=1}^{J} Z_i\right)^2\right] \le E\left[\left(\sum_{i=1}^{J} L\right)^2\right] = E\Big[J^2\Big]L^2, \tag{57}$$

since each of $Z_i \le L$ for all i. Let us look at the distribution of J. We notice that

$$\begin{aligned}\Pr(J = i) &= \Pr(J = i | \text{Island starts from first read})\\ &= \Pr(U_1 \le L, \ldots, U_{i-1} \le L, U_i > L)\\ &\le \Pr(U_1 \le L, \ldots, U_{i-1} \le L)\\ &\overset{(a)}{\le} \Pr(U_1 \le L) \Pr(U_2 \le L) \ldots \Pr(U_{i-1} \le L)\\ &\le \left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right) \times \cdots \times \left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right)\\ &= \left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right)^{i-1},\end{aligned}$$

where (a) is due to the fact that $U_i, i \in [1:K]$ are negatively associated. Now we can upper bound $E[J^2]$ as

$$\begin{aligned}E\Big[J^2\Big] &= \sum_{i=1}^{K} i^2 \Pr(J = i) \le \sum_{i=1}^{K} i^2 \left(1 - \left(1 - \frac{L}{n}\right)^{K-1}\right)^{i-1}\\ &\overset{(a)}{\le} \sum_{i=1}^{K} i^2 \left(1 - \exp\left(\frac{-\frac{LK}{n}}{\left(1 - \frac{L}{n}\right)}\right)\right)^{i-1}\\ &\overset{(b)}{\le} \sum_{i=1}^{K} i^2 \left(1 - \exp\left(-\frac{c}{\left(1 - \frac{\bar{L}\log 2}{2}\right)}\right)\right)^{i-1},\end{aligned}$$

where (a) is since $1 - x \ge e^{\frac{-x}{1-x}}$ and (b) is due to the max of $\log n/n$ being $\log 2/2$ for $n \in \mathbb{N}$.

The above series converges to a finite value. To see this, note that this summation is of the form $\sum_{i=1}^{K} i^2 \alpha^i$, for $\alpha \in (0, 1)$, which converges by the root test. Therefore $E[J^2]$ has to have

a finite bound (let this be M). This implies (57) can be upper bounded as

$$E[N_1^2] \leq M\bar{L^2}(\log n)^2.$$

This implies that $E[N_1^2/(\log n)^2]$ is bounded and finite.

APPENDIX H
PROOF OF EQUATION (25)

We aim to prove (25) here. First note that $E[K'']$ can be calculated as

$$\begin{aligned} E\big[K''\big] &= K\Pr(\text{Read has overlap size } \leq \delta \log n) \\ &= K\Pr(U_1 \geq L - \delta \log n) = K\left(1 - \frac{L - \delta \log n}{n}\right)^{K-1}. \end{aligned} \tag{58}$$

It is easy to see that $\lim_{n\to\infty} \frac{\log n}{n} E[K''] = \frac{c}{\bar{L}} e^{-c\sigma}$. Note that the formation of apparent islands can be interpreted as equivalently finding real islands with read lengths truncated by $\delta \log n$ (except for the last read in the island). The quantity $\lim_{n\to\infty} \frac{1}{n} E[K'']E[N_1^{\delta}]$ is just the coverage of this modified expression. But since the last read is still size L, we can think of that read in two parts, one part which contributes $L - \delta \log n$ and the other $\delta \log n$. Therefore the total coverage is

$$1 - e^{-c\sigma} + c(1-\sigma)e^{-c\sigma}. \tag{59}$$

This is because the reads of length $L - \delta \log n$ contribute to a coverage of $1 - e^{-c\sigma}$ (the effective coverage depth is shortened). Now the additional $\delta \log n$ contribute individually an extra length for each island, but this only needs to be added for the last read. Since there are K'' islands, the cumulative contribution is $\lim_{n\to\infty} \frac{1}{n} E[K''](\delta \log n) = c(1-\sigma)e^{-c\sigma}$.

Therefore from (58) and (59), we can say that

$$\begin{aligned} &\lim_{n\to\infty} \left(\frac{1}{n} E[K'']E[N_1] - E\big[K''\big] \frac{\log n}{n} \right) \\ &\quad = (1 - e^{-c\sigma}) + c(1-\sigma)e^{-c\sigma} - \frac{c}{\bar{L}} e^{-c\sigma}. \end{aligned} \tag{60}$$

REFERENCES

[1] 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.

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

[3] G. Bresler, M. Bresler, and D. Tse, "Optimal assembly for high throughput shotgun sequencing," *BMC Bioinformat.*, vol. 14, no. S5, p. S18, 2013.

[4] I. Shomorony, T. A. Courtade, and D. Tse, "Fundamental limits of genome assembly under an adversarial erasure model," *IEEE Trans. Mol. Biol. Multi-Scale Commun.*, vol. 2, no. 2, pp. 199–208, Dec. 2016.

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

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

[7] R. 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," *Angewandte Chemie Int. Edition*, vol. 54, no. 8, pp. 2552–2555, 2015.

[8] H. T. Yazdi, Y. Yuan, J. Ma, H. Zhao, and O. Milenkovic, "A rewritable, random-access DNA-based storage system," *Sci. Rep.*, vol. 5, Sep. 2015, Art. no. 14138.

[9] Y. Erlich and D. Zielinski, "DNA fountain enables a robust and efficient storage architecture," *Science*, vol. 355, p. 6328, pp. 950–954, 2017.

[10] L. Organick *et al.*, "Random access in large-scale DNA data storage," *Nat. Biotechnol.*, vol. 36, pp. 242–248, Feb. 2018.

[11] P. L. Antkowiak *et al.*, "Low cost DNA data storage using photolithographic synthesis and advanced information reconstruction and error correction," *Nat. Commun.*, vol. 11, p. 5345, Oct. 2020.

[12] 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)*, 2017, pp. 3130–3134.

[13] I. Shomorony and R. Heckel, "Capacity results for the noisy shuffling channel," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2019, pp. 762–766.

[14] I. Shomorony and R. Heckel, "DNA-based storage: Models and fundamental limits," *IEEE Trans. Inf. Theory*, vol. 67, no. 6, pp. 3675–3689, Jun. 2021.

[15] A. Lenz, P. H. Siegel, A. Wachter-Zeh, and E. Yaakobi, "Anchor-based correction of substitutions in indexed sets," 2019, *arXiv:1901.06840*.

[16] A. Lenz, P. H. Siegel, A. Wachter-Zeh, and E. Yaakobi, "Coding over sets for DNA storage," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2018, pp. 2411–2415.

[17] 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*, 2019, pp. 1–5.

[18] 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)*, 2020, pp. 8846–8850.

[19] 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.

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

[21] 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.

[22] I. Shomorony, S. H. Kim, T. A. Courtade, and D. N. Tse, "Information-optimal genome assembly via sparse read-overlap graphs," *Bioinformatics*, vol. 32, no. 17, pp. i494–i502, 2016.

[23] T. Holenstein, M. Mitzenmacher, R. Panigrahy, and U. Wieder, "Trace reconstruction with constant deletion probability and related results," in *Proc. SODA*, vol. 8, 2008, pp. 389–398.

[24] S. R. Srinivasavaradhan, M. Du, S. Diggavi, and C. Fragouli, "On maximum likelihood reconstruction over multiple deletion channels," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2018, pp. 436–440.

[25] M. Cheraghchi, J. Ribeiro, R. Gabrys, and O. Milenkovic, "Coded trace reconstruction," in *Proc. IEEE Inf. Theory Workshop (ITW)*, 2019, pp. 1–5.

[26] R. Gabrys and O. Milenkovic, "Unique reconstruction of coded sequences from multiset substring spectra," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2018, pp. 2540–2544.

[27] S. Marcovich and E. Yaakobi, "Reconstruction of strings from their substrings spectrum," 2019, *arXiv:1912.11108*.

[28] I. Shomorony and A. Vahid, "Communicating over the torn-paper channel," in *Proc. IEEE Global Commun. Conf.*, 2020, pp. 1–6.

[29] A. N. Ravi, A. Vahid, and I. Shomorony, "Capacity of the torn paper channel with lost pieces," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2021, pp. 1937–1942.

[30] I. Shomorony and A. Vahid, "Torn-paper coding," *IEEE Trans. Inf. Theory*, vol. 67, no. 12, pp. 7904–7913, Dec. 2021.

[31] S. Nassirpour and A. Vahid, "Embedded codes for reassembling non-overlapping random DNA fragments," *IEEE Trans. Mol. Biol. Multi-Scale Commun.*, vol. 7, no. 1, pp. 40–50, Mar. 2021.

[32] K. Joag-Dev and F. Proschan, "Negative association of random variables with applications," *Ann. Stat.*, vol. 11, no. 1, pp. 286–295, 1983.