Cover Your Bases: How to Minimize the Sequencing Coverage in DNA Storage Systems

Daniella Bar-Lev[*†], **Omer Sabary**[*†], **Ryan Gabrys**[‡], and **Eitan Yaakobi**[†]
[†]The Henry and Marilyn Faculty of Computer Science, Technion – Israel Institute of Technology, Haifa, Israel.
[‡]Calit2, University of California, San Diego.
Email: {daniellalev, omersabary, yaakobi}@cs.technion.ac.il, rgabrys@eng.ucsd.edu
[*]The two first authors contributed equally to this work.

***Abstract*—Although the expenses associated with DNA sequencing have been rapidly decreasing, the current cost stands at roughly $1.3K/TB, which is dramatically more expensive than reading from existing archival storage solutions today. In this work, we aim to reduce not only the cost but also the latency of DNA storage by studying the *DNA coverage depth problem*, which aims to reduce the required number of reads to retrieve information from the storage system. Under this framework, our main goal is to understand how to optimally pair an error-correcting code with a given retrieval algorithm to minimize the sequencing coverage depth, while guaranteeing retrieval of the information with high probability. Additionally, we study the DNA coverage depth problem under the random-access setup.**

I. Introduction

As a result of its remarkable density and durability, DNA is a promising storage medium. One of the main components in any DNA storage system [1], [8], [17], [23] is a DNA sequencer, which reads back the user's pre-stored information. Nowadays, DNA sequencers suffer from relatively slow throughput as well as high costs relative to other alternative storage technologies [19], [24], [25]. These issues are related to the so-called *coverage depth* of DNA storage, which is defined as the ratio between the number of reads that are sequenced and the number of synthesized oligos [12]. Reducing the coverage depth can improve the latency of any existing DNA storage system and reduce its costs.

Motivated by the connection between the coverage depth, latency, and cost, in this work we initiate the study of a novel problem, referred to as the *DNA coverage depth problem*. Simply stated, the DNA coverage depth problem aims to minimize the coverage depth while maintaining system reliability. In this work, we study the required coverage depth as a function of the DNA storage channel, the error-correcting code, and the reconstruction algorithm. Furthermore, we seek to understand how to pair an error-correcting code with a given reconstruction algorithm in order to minimize the coverage depth. This problem will be studied under both the random and non-random access settings.

The DNA coverage depth problem is related to the *coupon collector's (CCP)*, *dixie cup*, and *urn* problems [7], [9], [10], [16]. For all these problems, it is assumed that there are n different types of coupons and the question of interest is *how many coupons one should collect before possessing one coupon of each type*. It is well known that if the coupons are drawn uniformly at random (with repetition), then the expected number of coupons necessary to have at least one coupon from each type is roughly $n \log n$. Under our setting, the coupons refer to the copies of the synthesized oligos and the goal is to read at least one copy of every oligo.

The CCP has several generalizations [7], [10], [16], some of which will be explored in this work. One such problem, which is referred to as the *MDS coverage depth* problem studied in Section III, is *how many coupons one should collect before possessing t copies of k coupons*. This generalization represents the scenario where a reconstruction algorithm that requires t reads of an oligo for successful decoding is used along with an MDS code that requires retrieving k out of the n synthesized sequences to recover the stored encoded information. Our main result is a closed-form expression that upper bounds the expected number of coupons that need to be collected along with an upper bound that shows that in many cases the random variable of interest is, with high probability, below this upper bound. We also show that using the corresponding MDS code minimizes the expected number of coupons one needs to collect to retrieve the information.

Motivated by the random-access setting where one wishes to retrieve a single strand of DNA from a storage system, in Section IV we consider another problem that is related to the CCP, but to the best of our knowledge has not been studied before. Suppose we are given k information coupons which we can encode into a set of n total coupons. *For any information coupon say i, what is the expected number of coupons that need to be collected in order to retrieve the information in coupon i?* Trivially, if no code is used, then the expected number of coupons that need to be collected is equal to k. In Section IV, we initiate the study of this problem, which we refer to as the *singleton-random-access problem*. Our main result shows that it is indeed possible to design coding schemes that allow random access that requires less than k coupons and provide an example of such a scheme.

This paper is organized as follows. Section II, introduces the definitions that are used throughout the paper. Section III presents our results for the MDS coverage depth problem. We also show that in several instances MDS codes are optimal in the sense that they minimize the expected number of strands necessary for retrieval. Section IV introduces and presents our results for the singleton-random-access problem. Due to space limitations, some of the proofs can be viewed in the extended version of this paper [2].

II. Definitions and Channel Model

In the typical model of DNA-based storage systems [8], [17], [23], the data is stored as a codeword that can be described by a vector of length-ℓ *sequences* or *strands* over the alphabet $\Sigma = \{A, C, G, T\}$. In many cases an *outer error-*

The research was funded in part by the European Union (ERC, DNAStorage, 865630). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. This work was also supported in part by the Israel Innovation Authority grant 75855 and by NSF Grant CCF 2212437.

978-1-6654-7554-9/23/$31.00 ©2023 IEEE

correcting $\mathcal{C}$ is used to encode the data over the length-ℓ sequences, so it is assumed that the outer code $\mathcal{C}$ receives a vector of k length-ℓ sequences, $\mathcal{U} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k) \in (\Sigma^\ell)^k$ and returns a vector of n length-ℓ sequences $\mathcal{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n) \in (\Sigma^\ell)^n$. The vector $\mathcal{X}$ is the input to the DNA storage system which we now describe in more detail.

The DNA storage channel, denoted by S, first produces several noisy copies for each of the strands in $\mathcal{X}$. Then, these noisy copies are amplified using PCR, and lastly a *sample* of M of these strands is sequenced using a DNA sequencing technology [12]. Therefore, the output of the DNA storage channel can be described as a multiset $\mathcal{Y}_M = \{\!\{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_M\}\!\}$, where each $\boldsymbol{y}_j \in \Sigma^*$ for $j \in [M]$ is called a *read* and is a noisy version of some $\boldsymbol{x}_i$, $i \in [n]$, where $[n]$ denotes the set $\{1, \ldots, n\}$. The number of reads in $\mathcal{Y}_M$ that are noisy copies of the i-th strand $\boldsymbol{x}_i, i \in [n]$, depends upon some probability distribution $\boldsymbol{p} = (p_1, \ldots, p_n)$, where for $i \in [n]$, p_i is the probability to sample a read of $\boldsymbol{x}_i$. The probability distribution $\boldsymbol{p}$ is a function of the DNA storage channel S and is referred by the *channel probability distribution*, or in short *channel distribution*; Note that $\boldsymbol{p}$ might also depend on the design of the DNA strands in $\mathcal{X}$, however for simplicity, in this work we assume that $\boldsymbol{p}$ is only a function of the channel S.

Remark 1. Note that in several works; see e.g. [15], [21], it is assumed that a *set* (and not a vector) of strands is stored in the DNA storage system. However, since the strands in these sets are anyway tagged by indices, we assume for simplicity that the information is a vector of strands. Furthermore, it may also be possible that every strand is encoded using an inner code [8], [17]. Nevertheless, since this part is independent of our study in this work, it is not treated as part of the encoding step. Still, it is taken into account in the success probability of a retrieval algorithm, as will be explained next.

The decoding process of $\mathcal{X}$ (and thus $\mathcal{U}$) starts with partitioning the reads in $\mathcal{Y}_M$ into groups, also called *clusters*, according to their origin strand, i.e., for $i \in [n]$, the i-th cluster should contain all the reads $\boldsymbol{y}_j$ that are noisy copies of $\boldsymbol{x}_i$. To simplify the analysis, we assume that this step is accomplished error-free. In practice, this assumption can be reached using indices in the sequences which can be further protected using some error-correcting code [23]. Hence, the probability of successfully retrieving $\mathcal{X}, \mathcal{U}$ mainly depends on the following two components of the solution being used.

1) *Error-correcting code*. When $\mathcal{X}$ is a codeword in some error-correcting code $\mathcal{C}$, it is possible to successfully retrieve $\mathcal{X}$ even if not all of its n strands were decoded successfully. In particular, if $\mathcal{C}$ is an $[n, k]$ MDS code, then any k strands are sufficient to decode the data.
2) *The retrieval algorithm*. The success probability to retrieve the strand $\boldsymbol{x}_i$ also depends on the *retrieval algorithm*, which aims to decode a sequence using several of its noisy copies [3]. Typically, this probability depends on the number of noisy copies, the channel error rates, and the use of an inner code in the strands.

The main goal of this paper is to study the required sample size M that guarantees successful decoding of the information. According to our model, this sample size depends on the channel, the error-correcting code, and the channel probability distribution $\boldsymbol{p}$. We study two main problems in the following two sections. Section III covers the case where the goal is to retrieve the information vector $\mathcal{U}$, while in Section IV the goal is to retrieve a single information strand in $\mathcal{U}$, i.e., some $\boldsymbol{u}_i$, for $i \in [k]$. For these problems, we calculate the expected required sample size for noiseless/noisy channels and study how it can be minimized using coding schemes.

Remark 2. The analysis presented in this work relies on the assumption that the reads are received sequentially from the DNA storage channel. This is indeed the case when using Nanopore sequencing [22]. Moreover, we note that even in the case where the reads are obtained altogether, our results are relevant as we show that the required number of reads in the sequential case is with high probability below our bound.

Throughout this paper, we will use the notation $\log$ to denote natural logarithms with base e.

III. Coverage Depth in the DNA Storage Channel

This section studies the required sample size to retrieve the information vector $\mathcal{U}$ as a function of the DNA storage channel, the error-correcting code, and the retrieval algorithm. Under this framework, our goal is to understand how to optimally pair an error-correcting code with a given retrieval algorithm in order to minimize the sample size, while guaranteeing successful decoding with high probability.

The code $\mathcal{C}$ is denoted by (n, k) or $[n, k]$ to indicate that it is an MDS code. To simplify the analysis, it is further assumed that the retrieval algorithm is characterized by $t \in \mathbb{N}$ which indicates for every strand $\boldsymbol{x}_i$ the required minimum number of noisy reads that guarantee its successful retrieval. The decoding of $\mathcal{U}$ is successful when "enough unique strands" are successfully recovered by the retrieval algorithm (i.e., these are the strands with at least t reads). Here, enough refers to sets of unique strands which allow to decode the information vector $\mathcal{U}$. For example, for MDS codes, this refers to any k out of the n strands. According to this characterization, we let $\nu_t^{\boldsymbol{p}}(\mathcal{C})$ be the random variable that governs the number of reads that should be sampled for successful decoding of $\mathcal{U}$. When $\mathcal{C}$ is an $[n, k]$ MDS code, this notation is replaced by $\nu_t^{\boldsymbol{p}}(n, k)$. The uniform distribution is denoted by $\boldsymbol{p}_u \triangleq (\frac{1}{n}, \ldots, \frac{1}{n})$ and for brevity, we let $\nu_t(n, k) \triangleq \nu_t^{\boldsymbol{p}_u}(n, k)$. The two main problems of this section are defined below.

Problem 1. (**The MDS coverage depth problem.**) For given values of k and n, find the followings.

1) The expectation value $\mathbb{E}\left[\nu_t(n, k)\right]$.
2) The probability distribution of $\nu_t(n, k)$, i.e., for any $m \in \mathbb{N}$ find the value of $P[\nu_t(n, k) > m]$.

Problem 2. (**The coding coverage depth problem.**) For a given value of k, find the following.

1) Given n and $\boldsymbol{p}$, find an (n, k) code $\mathcal{C}$ that is optimal with respect to minimizing $\mathbb{E}\left[\nu_t^{\boldsymbol{p}}(\mathcal{C})\right]$.
2) The minimum value of $\mathbb{E}\left[\nu_t^{\boldsymbol{p}}(\mathcal{C})\right]$ over all possible codes $\mathcal{C}$ and channel distributions $\boldsymbol{p}$. That is, find the value $\mathsf{M}^{opt}(k) \triangleq \liminf_{\mathcal{C}, \boldsymbol{p}} \{\mathbb{E}\left[\nu_t^{\boldsymbol{p}}(\mathcal{C})\right]\}$.

This section first presents a survey of related work and known results. Next, Problem 2 is solved for the noiseless channel in Theorems 2 and 3. The solution to Problem 1.1 is in fact a known result and the expectation is given in (1).

However, we present in Theorem 5 a random variable, which is closely related to this value and is easier to compute, and a lower bound on its expectation. Furthermore, Theorem 4 shows an upper bound for the value in Problem 1.2, which also implies an upper bound for the value in Problem 1.1.

A. Related Work

For the noiseless channel, it is sufficient to have a single read of each $\boldsymbol{x}_i, i \in [n]$, to retrieve it. We note that if the channel distribution is the uniform distribution $\boldsymbol{p}_u$, and no code is defined on the data (i.e., $k = n$) then Problem 1.1 is equivalent to the classical *coupon collector's problem* [9]. This problem was first studied by Feller [9] where it was referred to as the *dixie cup problem*. Under the assumption that we have n coupons and it is equally likely to collect any coupon, the expected number of draws (i.e., sample size) required to get a single copy for each coupon is $\mathbb{E}[\nu_1(k = n, n)] = n \log n + \gamma n + \mathcal{O}(1)$, where $\gamma \sim 0.577$ is the Euler–Mascheroni constant. Furthermore, it was also proven [10] that $\mathbb{E}[\nu_1(n, k)] = n(H_n - H_{n-k})$, where H_n is the n-th harmonic number. It is well-known that when $n - k \to \infty$ and $\frac{n}{n-k} < \infty$, this expression is roughly $\mathbb{E}[\nu_1(n, k)] \approx n \log(n) - n \log(n - k) = n \log(\frac{n}{n-k})$.

For noisy channels, i.e., $t > 1$, the problem is closely related to the classical *urn problem* [7], [16]. Suppose there are n labeled urns and each can be filled with identical balls. At every round, a ball is thrown into one of the urns randomly. In each round the probability of throwing a ball to the j-th urn is denoted by p_j, for $1 \leq j \leq n$, and we let $\boldsymbol{p} = (p_1, \ldots, p_n)$. In [16], it was shown that in order to have t balls in each urn (or equivalently t copies per coupon), the expected sample size is $\mathbb{E}[\nu_t(k = n, n)] = n \log n + n(t-1) \log \log n + nC_t + o(n)$, where C_t is a constant that depends on t. Following that, Erdős and Rényi [7] proved that the distribution is tightly concentrated around the expectation. Flajolet et al. [10] generalized these results to a general discrete distribution on the coupons/balls and proved that the expected sample size to have at least t copies/balls for k out of the n coupons/urns is

$$\mathbb{E}[\nu_t^{\boldsymbol{p}}(n, k)] = \sum_{q=0}^{k-1} \int_0^{\infty} [u^q] \prod_{i=1}^{n} (e_{t-1}(p_i v) + u\,(e^{p_i v} - e_{t-1}(p_i v))) e^{-v} dv, \tag{1}$$

where $e_t(x) = \sum_{i=0}^{t} \frac{x^i}{i!}$ and for a polynomial $Q(u)$, $[u^q]Q(u)$ is the coefficient of u^q in $Q(u)$. This known result solves Problem 1.1, not only for $\boldsymbol{p}_u$ but for any channel distribution. As can be seen, for practical purposes, the expression in (1) is not easy to calculate. Hence, in Section III-C we solve a closely related problem and present a closed-form expression.

B. The Coding Coverage Depth Problem - Noiseless Channel

In this section, we focus on the setup where the channel is noiseless which refers to $t = 1$. Under this setup, the minimum sample size M is equivalent to the quantity which is governed by $\nu_1^{\boldsymbol{p}}(n, k)$. Our main result is to solve Problem 2 and to show that MDS codes are optimal for any channel distribution. Furthermore, we show that $\mathbb{E}[\nu_1^{\boldsymbol{p}}(n, k)]$ is minimized when $\boldsymbol{p}$ is the uniform distribution.

In light of the existing results and as a first step toward obtaining Theorem 3, we first show that for the uniform channel distribution, $\mathbb{E}[\nu_1(n, k)]$ decreases as n increases.

Claim 1. For all $n \geq k$, $\mathbb{E}[\nu_1(n, k)] > \mathbb{E}[\nu_1(n + 1, k)]$.

The next claim solves Problem 2.1 and states that given k information strands, for any channel distribution $\boldsymbol{p}$, using an $[n, k]$ MDS code results with the minimal expectation compared to any other length-n codes. This can be verified by showing that the number of subsets of size k sufficient to retrieve the information is maximized by MDS codes.

Claim 2. Given k, n and $\boldsymbol{p}$, and an (n, k) code $\mathcal{C}$ which is not an MDS code. It holds that, $\mathbb{E}[\nu_1^{\boldsymbol{p}}(n, k)] \leq \mathbb{E}[\nu_1^{\boldsymbol{p}}(\mathcal{C})]$.

The next theorem states that, when using an MDS code, $\mathbb{E}[\nu_1^{\boldsymbol{p}}(n, k)]$ is minimized when $\boldsymbol{p} = \boldsymbol{p}_u$.

Theorem 1. For any $\boldsymbol{p}$, $\mathbb{E}[\nu_1^{\boldsymbol{p}}(n, k)] \geq \mathbb{E}[\nu_1(n, k)]$.

Theorem 1, together with the above claims imply a lower bound on $\mathbb{E}[\nu_1^{\boldsymbol{p}}(n, k)]$, which is given next.

Theorem 2. For any channel distribution $\boldsymbol{p}$ it holds that, $\mathbb{E}[\nu_1^{\boldsymbol{p}}(n, k)] \geq \mathbb{E}[\nu_1(n, k)] = \sum_{i=0}^{k-1} \frac{n}{n-i} \approx n \log(\frac{n}{n-k})$.

Finally, we give the asymptotic value for $\mathbb{E}[\nu_1(n, k)]$.

Theorem 3. If $\frac{k}{n} = R$, for fixed $0 < R < 1$, we have that, $\lim_{n\to\infty} \frac{\mathbb{E}[\nu_1(n, k=nR)]}{k} = \frac{1}{R} \log\left(\frac{1}{1-R}\right)$. Otherwise, for any fixed k, it holds that, $\lim_{n\to\infty} \frac{\mathbb{E}[\nu_1(n, k=nR)]}{k} = 1$.

C. The MDS Coverage Depth Problem - Noisy Channel

In this section a noisy channel with uniform distribution is considered. Under this setup, we assume the data is encoded with an $[n, k]$ MDS code and that each strand $\boldsymbol{x}_i$ can be retrieved given $t > 1$ reads, which are noisy copies of it, and cannot be retrieved given less than t reads.

Our main result in this section is Theorem 4, in which an upper bound on the probability distribution of $\nu_{t+1}(k, n)$ (Problem 1.2) is given. This theorem is stated as follows.

Theorem 4. For n large enough and $\frac{r_x}{n} \geq t - 1$, we have

$$P\left[\nu_{t+1}(n, k) > r_x\right] \leq 1 - e^{-\frac{e^{-x}}{(t-1)!}}.$$

In particular, for any $\varepsilon > 0$, if at least r_x reads are sampled for $x = -\log((t-1)! \cdot \log((1-\varepsilon)^{-1}))$ and r_x as in (2), then the probability to fail to decode the information is at most ε.

The above theorem implies that when sampling more (or less) than r_0 reads, the upper bound on $P\left[\nu_{t+1}(n, k) > r_0\right]$ approaches zero (or one) according to a double exponential function. Thus, it can be said that the above probability is tightly concentrated below r_0 and hence we believe that r_0 is a closed-form expression that approximates $\mathbb{E}[\nu_t(n, k)]$.

To prove Theorem 4 we recall that within the context of the urn problem (see Section III-A), the random variable $\nu_{t+1}(n, k)$ denotes the number of balls (or rounds) necessary to guarantee that we have a set of k urns where each urn has at least $t + 1$ balls. For shorthand, for $x \in \mathbb{R}$, we let

$$r_x = n \log\left(\frac{n}{n-k}\right) + nt \log \log n + 2n \log(t + 1) + nx. \tag{2}$$

If r_x balls are drawn, then our main result, which is formally stated in Theorem 4, is to show that the probability that there are at least k urns each with at least $t+1$ balls is no more than $1 - e^{-\frac{e^{-x}}{(t-1)!}}$. Analogous to the approach used in the previous section, we will show that if the number of balls thrown is at least r_x, then the probability to have a most $n - k + 1$ urns which are *not* filled with $t + 1$ balls is upper bounded

by $1-e^{-\frac{e^{-x}}{(t-1)!}}$. The approach leveraged in this section is based on a technique first employed by Erdős and Rényi in [7] where they derived a concentration result for the classical urn problem that holds precisely when $n=k$. In the following, we show that a similar result holds for a more general setup. First, the following two related events are considered.

$E_{<,s}^{(r)}$**:** After r rounds, there exists a set $S_{<,s}$, of $n-k+1$ urns, each containing less than s balls. This event is our main interest in the section where $s=t+1$.

$E_{=}^{(r)}$**:** After r rounds, there exists a set $S_{=}$, of $n-k+1$ urns, each containing less than $t+1$ balls and there exists at least one urn in $S_{=}$ with exactly t balls.

At a high level, the proof works as follows. We first prove that $P(E_{<,t+1}^{(r)})-P(E_{=}^{(r)})\leq P(E_{<,t}^{(r)})$. Afterwards, we show that the probability of $E_{<,t}^{(r_x)}$ approaches zero when n is large. Thus, $P(E_{=}^{(r_x)})$ can be used to approximate $P(E_{<,t+1}^{(r_x)})$ when n is large. Finally, we show that $P(E_{=}^{(r_x)})\leq 1-e^{-\frac{e^{-x}}{(t-1)!}}$, which implies our main result in Theorem 4.

Claim 3. For any real x, $P(E_{<,t+1}^{(r_x)})-P(E_{=}^{(r_x)})\leq P(E_{<,t}^{(r_x)})$.

Claim 4. For n large enough and $\frac{r_x}{n}\geq t-1$, $P(E_{<,t}^{(r_x)})\to 0$.

The previous claim states that the probability that there are at least $n-k+1$ urns with less than t balls goes to zero as n grows. As a result of the previous two claims, we can bound the probability of $E_{<,t+1}^{(r_x)}$ by the probability the event $E_{=}^{(r_x)}$ occurs. To this end, we can work with the probability a simpler event occurs. We will consider the event that there is at least one urn (amongst all n) that contains exactly t balls in it. We denote this event by $A_t(n)$ and $P(E_{=})\leq P(A_t(n))$. The next claim was proved in [7].

Claim 5. For n large enough, $P(A_t(n))\leq 1-e^{-\frac{e^{-x}}{(t-1)!}}$.

Theorem 4 follows from Claim 3, Claim 4, and Claim 5.

As mentioned above, the solution to Problem 1.1 is given in (1). Nevertheless, the expression in (1) is not a closed-form expression, and thus it might be hard to compute it. For practical purposes of DNA storage systems, it is sometimes required to plan ahead and sample the number of reads that guarantees successful decoding. Hence, we turn to the following strongly related problem and give a closed-form expression to this related value. Turning back to the urn problem terminology, the goal is to find a lower bound on the number of rounds r, that guarantees that the expected number of urns that are *not filled* with $t+1$ balls is at most $n-k$. In order to derive this result, we first consider the probability that any fixed urn is *not filled* with $t+1$ or more balls by the r-th round. For $r\geq nt$, this probability is given by, $p=\sum_{j=0}^{t}\binom{r}{j}n^{-j}\left(1-\frac{1}{n}\right)^{r-j}\leq e^{-rD(\frac{t}{r}||\frac{1}{n})}$, where the last inequality follows from Chernoff bound [5] and $D(a||p)$ is the Kullback–Leibler divergence which is given by $D(a||p)\triangleq a\log_2\frac{a}{p}+(1-a)\log_2\frac{1-a}{1-p}$.

Under this setup, each of the n urns can be interpreted as a Bernoulli random variable with probability p, which is denoted by $X_i^{(r)}$ for $1\leq i\leq n$. Let $X^{(r)}\triangleq\sum_{i=1}^{n}X_i^{(r)}$ be the number of urns that are not filled with at least $t+1$ balls after r rounds, which implies that the number of urns that have at least $t+1$ balls is $n-X^{(r)}$. Our approach will be to determine a value for r, which guarantees (in expectation) that $X^{(r)}$ is at most $n-k$. From the linearity of expectation,

$$\mathbb{E}[X^{(r)}]=np\leq ne^{-t\log_2(\frac{nt}{r})-(r-t)\log_2\left(\frac{(r-t)n}{r(n-1)}\right)}. \tag{3}$$

The next claim will be used in the derivation to follow.

Claim 6. For $r\geq nt$, we have that $\mathbb{E}[X^{(r)}]\leq n-k$, if,

$$-\frac{r}{nt}e^{-\frac{r}{nt}}\geq -\frac{1}{e}\left(1-\frac{k}{n}\right)^{\frac{\log 2}{t}}. \tag{4}$$

The values of r for which (4) holds can be deduced using the Lambert W function [6, Section IV], [4, Theorem 1].

Theorem 5. Let $R=\frac{k}{n}$. For any $r\geq nt+n\log 2\log(1-R)+nt\sqrt{-\frac{2\log 2}{t}\log(1-R)}$, we have that $\mathbb{E}[X^{(r)}]\leq n-k$.

Practically speaking, as mentioned above, the noisy channel fits the real scenario of DNA storage systems. Hence, it should be mentioned that a similar problem was studied experimentally by Erlich and Zielinski [8], however, with a slightly different setup. They presented the DNA fountain, a Luby transform-based scheme and assumed that the total number of reads is fixed and given (from the DNA sequencer) and it is distributed with a negative binomial distribution. Thus, they were able to calculate the average number of copies per strand and empirically evaluate the required sample size as a function of the distribution's parameters. It should be noted that they only considered reads of the design length and thus the error rates were reduced. They also evaluated how dilution affects the distribution and the required sample size.

Finally, another variation of the noisy channel $\mathcal{S}$ is studied, which is relevant to the DNA fountain [8] and similar schemes. Here, it is required to obtain a single noiseless copy from k out of the n synthesized strands. Assuming uniform distribution on the strands, in this channel, any sampled read is drawn noiseless with some probability $0<\alpha<1$. We use the notation of $\omega_\alpha(n,k)$ to denote the random variable describing the required sample size to ensure successful decoding of this case. This case is easier to analyze, and the following results can be derived using similar techniques to the one used to solve the classical coupons collector's problem [9].

Theorem 6. For any $k\leq n$, $\mathbb{E}[\omega_\alpha(n,k)]=\frac{n}{\alpha}\left(H_n-H_{n-k}\right)$.

IV. Random Access

In this section we define the problem of optimizing the sample size for random access queries in DNA storage systems. In this problem, a vector of k information strands, each of length ℓ, $\mathcal{U}=(\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots,\boldsymbol{u}_k)\in(\Sigma^\ell)^k$, is encoded into a vector of n strands of length ℓ, $\mathcal{X}=(\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_n)\in(\Sigma^\ell)^n$ that are stored in the DNA storage channel as described in Section II. Later, the user wishes to retrieve a subset of the k information strands. In this work, we consider the special case in which this subset is a singleton, i.e., the case where the user wishes to retrieve a single information strand $\boldsymbol{u}_i$ for some $i\in[k]$. More formally, we are interested in the following problem.

Problem 3 (The singleton coverage depth problem). Given an (n,k) code $\mathcal{C}$, for $i\in[k]$, let $\tau_i(\mathcal{C})$ be the random variable that denotes the number of samples to recover the i-th information strand. Find the following:

1) The expectation value $\mathbb{E}[\tau_i(\mathcal{C})]$ and the probability distribution $P[\tau_i(\mathcal{C})>r]$ for any $r\in\mathbb{N}$.

2) The maximal expected number of samples to retrieve an information strand, i.e., $T^{\mathcal{C}}_{\max} \triangleq \max_{1\le i\le k} \mathbb{E}[\tau_i(\mathcal{C})]$.

When no coding is used, $\mathcal{C}$ is removed from the notations.

The next claim solves Problem 3 when no coding is used.

Lemma 1. Let $n \ge 1$. For any $1 \le i \le n$, we have that $\mathbb{E}[\tau_i] = n$ and $T_{\max} = n$. Additionally, for any $r \in \mathbb{N}$ we have that $P[\tau_i > r] = \left(1 - \frac{1}{n}\right)^r$, and $P[\tau_i = r] = \frac{1}{n} \cdot \left(1 - \frac{1}{n}\right)^{r-1}$.

Before we continue to more involved cases, we define the n random variables $\widehat{\tau}_i(\mathcal{C}), i \in [n]$, such that $\widehat{\tau}_i(\mathcal{C})$ governs the required sample size to retrieve the i-th encoded strand. Additionally, for every set $J \subseteq [n]$, let $\widehat{\tau}_J(\mathcal{C}) \triangleq \max_{i\in J} \widehat{\tau}_i(\mathcal{C})$. These random variables are used as a technical tool in our analysis and the key idea is given in the next lemma.

Claim 7. For any (n, k) code $\mathcal{C}$ and any $J \subseteq [n]$ of size ρ we have that $\mathbb{E}[\widehat{\tau}_J] = nH_\rho$.

The structure of $\mathcal{C}$ defines for each information strand all possible sets of encoded strands that are sufficient for its recovery. This concept is similar to recovery sets in *locally repairable codes* [20] as well as the ones with *availability* [14].

Definition 1. Let $\mathcal{C}$ be an (n, k) code. We say that $J \subseteq [n]$ is a *retrieval set* of the i-th information strand (i.e., $\boldsymbol{u}_i$) if it is possible to decode the information strand $\boldsymbol{u}_i$ from the encoded strands whose indices belong to J. The set of all retrieval sets of $\boldsymbol{u}_i$ is denoted by $\widehat{\mathcal{D}}(i)$, and $\mathcal{D}(i)$ is the set of all minimal retrieval sets of $\boldsymbol{u}_i$ (with respect to inclusion relation).

Next, we consider the case of non-systematic codes for $k = n$ (i.e., $\mathcal{U} \neq \mathcal{X}$). Since $\mathcal{X}$,$\mathcal{U}$ have the same length, given any set of strands $\{\boldsymbol{x}_i : i \in J\}$, we can recover at most $|J|$ information strands from $\mathcal{U}$. Our goal is to extend Lemma 1 to the coded case when $k = n$ using this basic insight.

Claim 8. For any $(n = k, k)$ code $\mathcal{C}$, we have that $T^{\mathcal{C}}_{\max} \ge T_{\max} = n$. In particular, if we let ρ_i be the size of the smallest retrieval set for the information strand $\boldsymbol{u}_i$, then $\mathbb{E}[\tau_i(\mathcal{C})] = nH_{\rho_i}$ and $T^{\mathcal{C}}_{\max} = nH_\rho$, where $\rho \triangleq \max_i \rho_i$.

Proof: If each $\boldsymbol{u}_i$ can be retrieved from a single strand $\boldsymbol{x}_j$, then we have that $T^{\mathcal{C}}_{\max} = T_{\max} = n$. Otherwise, assume w.l.o.g. that $\boldsymbol{u}_1$ can not be retrieved from a single strand and let $J \subseteq [n]$ be a set of minimal size $|J| = \rho_1$ such that $J \in \mathcal{D}(1)$. Since $k = n$, $|\mathcal{D}(1)| = 1$, i.e. the set J is unique and hence by Claim 7, $\mathbb{E}[\tau_i(\mathcal{C})] = \mathbb{E}[\widehat{\tau}_J(\mathcal{C})] = nH_{\rho_1} > n$, where the last inequality holds since $|J| = \rho_1 > 1$. Thus,

$$T^{\mathcal{C}}_{\max} = \max_{1\le i\le k} \mathbb{E}[\tau_i(\mathcal{C})] = \max_{1\le i\le k} nH_{\rho_i} = nH_\rho.$$

■

We continue by studying cases where $n > k$. Next, the case where the minimal retrieval sets are disjoint is considered.

Theorem 7. Let $\mathcal{C}$ be an (n, k) code and $i \in [k]$. If $\mathcal{D}(i) = \{A, B\}$ for two disjoint retrieval sets, i.e. $A \cap B = \emptyset$, then $\mathbb{E}[\tau_i(\mathcal{C})] = n \cdot \left(H_{|A|} + H_{|B|} - H_{|A|+|B|}\right)$.

The proof of Theorem 7 relies on the inclusion-exclusion principle and can be extended to more than two retrieval sets.

Corollary 1. For $i \in [k]$, if $\mathcal{D}(i) = \{A_1, A_2, \ldots, A_v\}$ for mutually disjoint retrieval sets, then

$$\mathbb{E}[\tau_i(\mathcal{C})] = n \cdot \sum_{s=1}^{v} (-1)^{s+1} \sum_{1\le j_1<\cdots<j_s\le v} H_{(|A_{j_1}|+\cdots+|A_{j_s}|)}.$$

An additional conclusion from Theorem 7 is given below.

Corollary 2. Assume $\mathcal{C}$ is the $(n = k + 1, k)$ simple parity code (i.e., $\mathcal{X} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k, \sum_{j=1}^{k} \boldsymbol{u}_j)$. Then, for any $i \in [k]$, we have that, $\mathbb{E}[\tau_i(\mathcal{C})] = k$ and $T^{\mathcal{C}}_{\max} = k$.

Corollary 2 states that the simple parity code does not improve the value of $T^{\mathcal{C}}_{\max}$. This observation raises the problem of finding codes that indeed improve this parameter, and next we consider MDS codes for this purpose. First recall that by Lemma 1, if no code is used, then we have that $T_{\max} = \mathbb{E}[\tau_i] = k$ for any $i \in [k]$. On the other hand, if a non-systematic $[n, k]$ MDS code is used, then to retrieve any specific information strand, one should sample a subset of k distinct encoded strands. Hence, by Theorem 2, for any $i \in [k]$, we have that, $T^{\mathcal{C}}_{\max} \ge \mathbb{E}[\tau_i(\mathcal{C})] = \sum_{i=0}^{k-1} \frac{n}{n-i} \approx n \log(\frac{n}{n-k})$, while by Theorem 3, if $\frac{k}{n} = R$, for fixed $0 < R < 1$ it follows that $T^{\mathcal{C}}_{\max} \ge \mathbb{E}[\tau_i(\mathcal{C})] > k$. Next, the case where $\mathcal{C}$ is a systematic MDS code is analyzed.

Theorem 8. Let $\mathcal{C}$ be a systematic $[n, k]$ MDS code and assume that $k > \frac{n}{2}$. For any $i \in [k]$ we have that $\mathbb{E}[\tau_i(\mathcal{C})] = k$.

In all the codes we studied so far, the expected number of reads to retrieve a single information strand $\boldsymbol{u}_i$, was at least k, which means that these code do not improve upon the case where no coding is used. Next, we give an example for a code $\mathcal{C}$ that can achieve $T^{\mathcal{C}}_{\max} < k$.

Example 1. Let $\mathcal{U} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3, \boldsymbol{u}_4) \in (\Sigma^\ell)^4$ and let

$$\mathcal{X} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3, \boldsymbol{u}_4, \boldsymbol{u}_1 + \boldsymbol{u}_2, \boldsymbol{u}_2 + \boldsymbol{u}_3, \boldsymbol{u}_3 + \boldsymbol{u}_4, \boldsymbol{u}_4 + \boldsymbol{u}_1).$$

Denote $\boldsymbol{x}_{i,j} \triangleq \boldsymbol{u}_i + \boldsymbol{u}_j$ and w.l.o.g. assume that we are interested in retrieving $\boldsymbol{u}_1$. It can be verified that

$$\mathcal{D}(1) = \left\{ \begin{matrix} \{\boldsymbol{u}_1\}, \ \{\boldsymbol{u}_2, \boldsymbol{x}_{1,2}\}, \ \{\boldsymbol{u}_4, \boldsymbol{x}_{1,4}\}, \\ \{\boldsymbol{u}_3, \boldsymbol{x}_{2,3}, \boldsymbol{x}_{1,2}\}, \ \{\boldsymbol{u}_3, \boldsymbol{x}_{3,4}, \boldsymbol{x}_{1,4}\}, \\ \{\boldsymbol{u}_4, \boldsymbol{x}_{3,4}, \boldsymbol{x}_{2,3}, \boldsymbol{x}_{1,2}\}, \ \{\boldsymbol{u}_2, \boldsymbol{x}_{3,4}, \boldsymbol{x}_{2,3}, \boldsymbol{x}_{1,4}\} \end{matrix} \right\},$$

while $\mathcal{D}(1)$ is given with an abuse of notation, in which the retrieval sets are given in terms of the encoded strands rather than their indices. Let $\mathcal{E}_{r-1}$ be the number of unique strands that were sampled in the first $r - 1$ draws. Since any set of 6 or more unique strands is a retrieval set of $\boldsymbol{u}_1$, we have that

$$P[\tau_1(\mathcal{C}) \ge r] = \sum_{i=1}^{5} P[\tau_1(\mathcal{C}) \ge r | \mathcal{E}_{r-1} = i] \cdot P[\mathcal{E}_{r-1} = i].$$

It can be verified that $P[\tau_1(\mathcal{C}) \ge r | \mathcal{E}_{r-1} = 1] = \frac{7}{8}$. In case $\mathcal{E}_{r-1} = 2$, there are $\binom{8}{2} = 28$ different pairs of strands, and since $\tau_1(\mathcal{C}) \ge r$, we should consider only the pairs from which $\boldsymbol{u}_1$ can not be retrieved. Hence we have that $P[\tau_1(\mathcal{C}) \ge r | \mathcal{E}_{r-1} = 2] = \frac{19}{28}$. Similarly, it can be verified that $P[\tau_1(\mathcal{C}) \ge r | \mathcal{E}_{r-1} = 3] = \frac{23}{56}$, $P[\tau_1(\mathcal{C}) \ge r | \mathcal{E}_{r-1} = 4] = \frac{8}{70}$, and $P[\tau_1(\mathcal{C}) \ge r | \mathcal{E}_{r-1} = 5] = \frac{1}{56}$. Furthermore, using the inclusion-exclusion principle, it can be proved that $P[\mathcal{E}_{r-1} = i] = \frac{\binom{8}{i}}{8^{r-1}} \sum_{j=0}^{i-1} \binom{i}{j} (-1)^j (i - j)^{r-1}$. By combining all of the above together we obtain that

$$\mathbb{E}[\tau_1(\mathcal{C})] = \sum_{r=1}^{\infty} P\left[\mathcal{T}_1^{\mathcal{C}} \ge r\right] = \frac{403}{105} \approx 3.838.$$

This section concludes with a lower bound on $\mathbb{E}[\tau_i(\mathcal{C})]$.

Lemma 2. For any (n, k) code $\mathcal{C}$, $T^{\mathcal{C}}_{\max} \ge \frac{k+1}{2}$.

REFERENCES

[1] L. Anavy, I. Vaknin, O. Atar, R. Amit, and Z. Yakhini, "Data storage in DNA with fewer synthesis cycles using composite DNA letters," *Nature Biotechnology* vol. 37, no. 1237, 2019.

[2] D. Bar-Lev, O. Sabary, R. Gabrys, and E. Yaakobi. "Cover your bases: How to minimize the sequencing coverage in DNA storage systems," *arXiv*, arxiv.org/abs/2305.05656, 2023.

[3] V. Bhardwaj, P. A. Pevzner, C. Rashtchian , and Y. Safonova, "Trace Reconstruction Problems in Computational Biology," *IEEE Trans. on Information Theory*, vol. 67, no. 6, 2021.

[4] I. Chatzigeorgiou, "Bounds on the Lambert Function and Their Application to the Outage Analysis of User Cooperation," *IEEE Communications Letters*, vol. 17, pp. 1505–1508, 2013.

[5] H. Chernoff, "A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on the Sum of Observations," *The Annals of Mathematical Statistics*, vol. 23, no. 4, pp. 493–507, 1952.

[6] R. Corless, H. Gonnet, D. Hare, D. J. Jeffrey, and D. E. Knuth, "On the LambertW function," *Advances in Computational Mathematics*, vol. 5, pp. 329–359, 1996.

[7] P. Erdős, and A. Rényi, "On a classical problem of probability theory," *Magyar Tud. Akad. Mat. Kutató Int.* vol. 6, no. 1-2, pp. 215–220, 1961.

[8] Y. Erlich, and D. Zielinski, "DNA Fountain enables a robust and efficient storage architecture," *Science*, vol. 335, no. 6328, pp. 950-954, 2017.

[9] W. Feller, "An introduction to probability theory and its applications," *Wiley*, vol. 1, 2nd edition, 1967.

[10] P. Flajolet, D. Gardy, and L. Thimonier, "Birthday paradox, coupon collectors, caching algorithms and self-organizing search," *Discrete Applied Mathematics*, vol. 39, no. 3, pp. 207-229, 1992.

[11] F. E. Harris, "Chapter 9 - gamma function," in *Mathematics for Physical Science and Engineering, Academic Press*, pp. 325–347, https://www.sciencedirect.com/topics/mathematics/digamma-function ,2014.

[12] R. Heckel, G. Mikutis, and R. N. Grass, "A Characterization of the DNA Data Storage Channel," *Scientific Reports*, vol. 9, no. 9663, 2019.

[13] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *The collected works of Wassily Hoeffding*, pp. 409–426, 1994.

[14] P. Huang, E. Yaakobi, H. Uchikawa and P. H. Siegel, "Linear locally repairable codes with availability," *IEEE International Symposium on Information Theory (ISIT)*, pp. 1871-1875, 2015.

[15] A. Lenz, P. H. Siegel, A. Wachter-Zeh, and E. Yaakobi, "Coding Over Sets for DNA Storage," *IEEE Transactions on Information Theory*, vol. 66, no. 4, pp. 2331–2351, 2020.

[16] D. J. Newman, "The Double Dixie Cup Problem," *The American Mathematical Monthly*, vol. 67, no. 1, pp. 58-61, 1960.

[17] L. Organick, S.D. Ang, Y. J. Chen , R. Lopez, S. Yekhanin, K. Makarychev, M. Z. Racz, G. Kamath, P. Gopalan, B. Nguyen, C. N. Takahashi, S. Newman, H. Y. Parker, C. Rashtchian, K. Stewart, G. Gupta, R. Carlson, J. Mulligan, D. Carmean, G. Seelig, L. Ceze, K. Strauss, "Random access in large-scale DNA data storage," *Nature Biotechnology*, vol. 36, no. 3, pp. 242–248, 2018.

[18] A. N. Philippou, C. Georghiou, G. N. Philippou, "A generalized geometric distribution and some of its properties," *Statistics & Probability Letters*, vol. 1, no. 4, pp.171–175, 1983.

[19] I. Shomorony and R. Heckel, "Information-Theoretic Foundations of DNA Data Storage," *Foundations and Trends in Communications and Information Theory* vol. 19, no. 1, pp 1–106, 2022.

[20] D. S. Papailiopoulos, and A. G. Dimakis, "Locally Repairable Codes," *IEEE Transaction on Information Theory*, vol. 60, no. 10, 2014.

[21] J. Sima, N. Raviv, and J. Bruck, "On Coding Over Sliced Information," *IEEE International Symposium on Information Theory (ISIT)*, pp. 767–771, 2019.

[22] Y. Wang, Y. Zhao, A. Bollas, Y. Wang, and K. F. Au, "Nanopore sequencing technology, bioinformatics and applications," *Nature Biotechnology*, no. 39, pp. 1348–1365, 2021.

[23] S.M.H.T. Yazdi, R. Gabrys, and O. Milenkovic, "Portable and Error-Free DNA-Based Data Storage," *Scientific Reports* vol. 7, no. 5011, 2017.

[24] S. M. H. T. Yazdi, H. M. Kiah, E. Garcia-Ruiz, J. Ma, H. Zhao, and O. Milenkovic, "DNA-Based Storage: Trends and Methods," *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, vol. 1, no. 3, pp. 230–248, 2015

[25] White paper by DNA Data Storage Alliance, "Preserving Our Digital Legacy: an Introduction to DNA Data Storage," a publication of *DNA Data Storage Aliance*, 2021.