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

Daniella Bar-Lev, *Student Member, IEEE,* **Omer Sabary**, *Student Member, IEEE,*
Ryan Gabrys, *Senior Member, IEEE,* and **Eitan Yaakobi**, *Senior Member, IEEE*

Abstract—Although the expenses associated with DNA sequencing have been rapidly decreasing, the current cost of sequencing information stands at roughly \$120/GB, 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 initiating the study of 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 the effect of error-correcting codes and retrieval algorithms on the required sequencing coverage depth. We establish that the expected number of reads that are required for information retrieval is minimized when the channel follows a uniform distribution. We also derive upper and lower bounds on the probability distribution of this number of required reads and provide a comprehensive upper and lower bound on its expected value. We further prove that for a noiseless channel and uniform distribution, MDS codes are optimal in terms of minimizing the expected number of reads. Additionally, we study the DNA coverage depth problem under the random-access setup, in which the user aims to retrieve just a specific information unit from the entire DNA storage system. We prove that the expected retrieval time is at least k for $[n, k]$ MDS codes as well as for other families of codes. Furthermore, we present explicit code constructions that achieve expected retrieval times below k and evaluate their performance through analytical methods and simulations. Lastly, we provide lower bounds on the maximum expected retrieval time. Our findings offer valuable insights for reducing the cost and latency of DNA storage.

I. INTRODUCTION

The world's digital data is growing exponentially, doubling from 30 to 64 zettabytes in just three years, and it is anticipated to reach 180 zettabytes by 2025, resulting in many new data storage challenges. The demand for storage capacity already exceeds the supply, and the gap continues to grow [32]. Recent research and insights from the IDC emphasize the struggle of existing storage technologies to meet the demands of the big data era.

Recognizing this challenge, DNA emerges as a promising storage medium due to its exceptional density and durability. The DNA storage pipeline usually involves three main components. The first is *DNA synthesis*, which produces artificial DNA molecules. These synthetic DNA molecules are called *oligos* or *strands* and they can be designed in a way that encodes the user's information. The current synthesis technologies only produce strands that are up to a length of 300 bases [24] and due to technology limitations, they also produce several noisy copies per encoded strand. Thus, it is likely that the user information is stored in several different strands. The second component of the DNA storage pipeline is a *storage container*, usually a small tube that contains all the short strands that encode the user information. Lastly, to read back the user information, it is required to perform *DNA sequencing* on the strands in the tube. The sequencing process translates the DNA strands into digital sequences over the DNA alphabet, which are noisy copies of the synthesized strands. These DNA sequences can be decoded to read back the user's information. In DNA storage, the retrieval time of the information can be measured as the time between the beginning of the sequencing process and the end of the system's decoding procedure.

The sequencing process, which is done using a DNA sequencer, is one of the principal components in any DNA storage system [2], [13], [26], [40]. Nowadays, DNA sequencers suffer from relatively slow throughput as well as high costs relative to other alternative storage technologies [35], [41], [43]. 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 designed strands [19]. As the retrieval time depends on the specific technology that is being used, in this work we consider the number of reads that should be sequenced instead of the actual retrieval time[1]. This simplification can be done since once a specific sequencing technology and algorithms are selected, the retrieval mainly depends on the number of reads that should be sequenced [2], [13], [26]. Moreover, sequencing costs depend on the number of reads, so 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, and in an effort to design coding schemes that overcome the drawbacks associated with existing sequencing technologies, in this work we initiate the study

---

[1]Note that the number of reads that should be sequenced is directly proportional to the coverage depth.

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. We will study the required coverage depth as a function of the DNA storage channel, the error-correcting code, and the algorithms involved in retrieving the user's information. Furthermore, we seek to understand how to pair an error-correcting code with a given DNA storage system in order to minimize the coverage depth. This problem will be studied under both the random and non-random access settings. While the latter addresses the problem of retrieving all the information that was being stored, the former describes the case in which the user is interested in retrieving only a specific part of the stored information. Moreover, we plan to suggest coding schemes that optimize the required coverage depth and to study, both theoretically and experimentally, how one can utilize codes to minimize the sequencing time and costs.

Despite significant work on DNA storage, only a small number of works have focused on reducing the latency and costs associated with sequencing in experimental or theoretical setups. Erlich et. al. [13] encoded digital information into DNA strands using a Luby transform-based coding scheme. Later, they diluted their synthesized strands and studied the effect of this dilution on their ability to sequence and decode the information. The dilution procedure reduced the potential (maximal) coverage depth of their system down to roughly 1300 reads per strand, thus making the decoding process more challenging. They showed that thanks to the error-correcting capability of their scheme, they were able to perfectly retrieve the stored information. In another related work, Chandak et. al. [7] defined the ratio between the number of synthesized bits and the number of information bits as the *writing cost*, and similarly the ratio between the number of bits that have to be read (sequenced) and the number of information bits was defined as the *reading cost*. In their work, they studied the tradeoffs and relations between the writing and reading costs. They first showed that for the noiseless channel, it is enough to read one copy per designed strand. Thus, the relation of these two costs can be obtained by inferring the channel as an erasure channel with an erasure probability that can be approximated using Poisson approximation. Additionally, the authors suggested an LDPC-based coding scheme that can improve the ratio between the two costs. They also showed by simulations how their suggested scheme can be used with different redundancy levels to reduce both the writing cost and the reading cost.

The DNA coverage depth problem is related to the *coupon collector's (CCP)*, *dixie cup*, and *urn* problems [12], [15], [16], [25]. 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 [12], [16], [25], some of which will be explored in this work. One such problem, which is referred to as the *MDS coverage depth* problem, 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 correctly retrieving k out of the n synthesized sequences to recover the stored encoded information. Another problem that is addressed in this paper is the *coding coverage depth problem*, which generalizes the MDS coverage depth and considers the effect of an error-correcting code, which is not necessarily an MDS code. Under this setup, our main results show that MDS codes are optimal codes for the purpose of reducing the expected coverage depth. Furthermore, our analysis for the MDS coverage depth problem provides a deep understanding of the required number of reads that should be sequenced in order to guarantee a successful retrieval of the information with high probability.

Additionally, motivated by the random-access setting where one wishes to retrieve a single strand of DNA from a storage system, in Section VI 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 and every coupon is collected with the same probability, then the expected number of coupons that need to be collected is equal to k. In Section VI, we initiate the study of this problem, which we refer to as the *singleton-random-access problem*. Our main result is to show that it is indeed possible to design coding schemes that allow random access that requires less than k coupons and provide examples of several such schemes.

This paper is organized as follows. Section II introduces the definitions that are used throughout the paper. In Section III, we formally define the problems that are studied throughout this paper along with related work. Section III also gives a detailed summary of the main results presented in this paper. Next, in Section IV, we consider the MDS coverage depth problem, where Section IV-A is devoted to the case where the channel is noiseless. Section IV-B extends the study of the MDS coverage depth problem to noisy channels, and gives several bounds on the success probability of the decoding as a function of the number of reads that were sequenced. Following that, in Section V, we study the coding coverage depth problem for the case where the channel is noiseless. The singleton-random-access problem is studied in Section VI. Finally, Section VII concludes the paper and gives a few directions for future research. For a detailed summary of the results and contributions presented in each of the sections, the reader is referred to Section III-C.

II. DEFINITIONS AND CHANNEL MODEL

In the typical model of DNA-based storage systems [13], [26], [40], the data is stored as a codeword that can be

described by a vector of length-ℓ *sequences* or *strands* over the alphabet $\Sigma = \{A, C, G, T\}$. The set of all length-ℓ vectors over Σ is denoted by Σ^ℓ, and $\Sigma^* \triangleq \bigcup_{\ell=0}^{\infty} \Sigma^\ell$. For a positive integer n, $[n]$ denotes the set $\{1, \ldots, n\}$. In many cases an *outer error-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, $\mathbf{U} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k) \in (\Sigma^\ell)^k$ and returns a vector of n length-ℓ sequences $\mathbf{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n) \in (\Sigma^\ell)^n$. For two vectors $\mathbf{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k_1})$ and $\mathbf{V} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{k_2})$, we denote by $\mathbf{U} \circ \mathbf{V}$ their *concatenated vector*, i.e., $\mathbf{U} \circ \mathbf{V} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k_1}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_{k_2})$. In this work, the code $\mathcal{C}$ is denoted by (n, k) or by $[n, k]$ in case $\mathcal{C}$ is an MDS code. The vector $\mathbf{X}$ is the input to the DNA storage system, which we now describe in more detail and is also illustrated in Fig. 1.

The DNA storage channel, denoted by S, first produces many noisy copies for each of the strands in the vector $\mathbf{X}$. Then, these noisy copies are amplified using Polymerase Chain Reaction (PCR), and lastly, a *sample* of M of these strands is sequenced using a DNA sequencing technology [19]. 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 $\boldsymbol{x}_i$, $i \in [n]$. It should be noted that our model assumes that for any read $\boldsymbol{y}_j$, the index $i \in [n]$ such that $\boldsymbol{y}_j$ is a noisy copy of $\boldsymbol{x}_i$ is known (this can be achieved by encoding the index i within the strand $\boldsymbol{x}_i$, e.g., as its prefix). Depending on the specific sequencing technology being used, the reads in $\mathcal{Y}_M$ can be obtained either sequentially (one after the other), or altogether. The former corresponds to Nanopore sequencing technologies [38], while the latter describes next-generation sequencing (NGS) technologies [6] (e.g. Illumina). 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 categorical 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 the distribution $\boldsymbol{p}$ might also depend on the design of the DNA strands in $\mathbf{X}$, however for simplicity, in this work we assume that $\boldsymbol{p}$ is only a function of the channel S. Moreover, it is assumed that $p_i > 0$ for all $i \in [n]$, unless stated otherwise.

Remark 1. Note that in several works, see e.g. [23], [34], 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 tagged by indices anyway, 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 [13], [26]. Nevertheless, since this part is independent of the study of this work, it is not treated as part of the encoding process, but it is taken into account in the success probability of a retrieval algorithm, as will be explained below.

The decoding process of $\mathbf{X}$ (and thus $\mathbf{U}$) 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 sequence $\boldsymbol{x}_i$ which can be further protected using some error-correcting code [40]. Hence, the probability of successfully retrieving $\mathbf{X}$ and $\mathbf{U}$ mainly depends on the following two components of the solution being used.

1) *Error-correcting code*. When $\mathbf{X}$ is a codeword in some error-correcting code $\mathcal{C}$, it is possible to successfully retrieve $\mathbf{X}$ even if not all of its n symbols were decoded successfully. The applicable subsets $J \subseteq [n]$ such that $\mathbf{X}$ can be retrieved from the symbols $\boldsymbol{x}_j$ for $j \in J$ are determined by the code $\mathcal{C}$. For example, if $\mathcal{C}$ is an $[n, k]$ MDS code, then any k strands (symbols of $\mathbf{X}$) 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 noisy copies [5]. Typically, this probability depends on the number of noisy copies which are given as input, the channel error rates, and the use of an inner code within the strands. In this work, we model the retrieval algorithm using an integer $t \geq 1$, and we assume that each strand $\boldsymbol{x}_i$ can be retrieved given t reads, which are noisy copies of it, and cannot be retrieved given less than t reads[2].

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 $\boldsymbol{p}$.

Remark 2. The analysis presented in this work assumes that the reads in the multiset $\mathcal{Y}_M$ are received sequentially from the DNA storage channel as illustrated in step 5a of Fig. 1. However, our results are also relevant for the case in which all the reads are obtained together. More specifically, the random variable that governs the sample size M for which decoding is possible in the sequential case can be used to describe the non-sequential case as well. That is, the probability distribution of the latter corresponds to the decoding success probability given M strands in the non-sequential case.

In this paper, we explore two different scenarios concerning our problem. In the first scenario, discussed in both Section IV and Section V, we focus on the objective of recovering all the stored information. This involves retrieving the entire vector $\mathbf{U}$. On the other hand, in Section VI, we shift our attention to a different scenario where our goal is to retrieve a specific part of the information, i.e., a specific subset of symbols from the vector $\mathbf{U}$. For these scenarios, we calculate the expected required sample size for noiseless/noisy channels and study how it can be minimized using coding schemes.

---

[2]Note that, in practice, the probability that the retrieval algorithm succeeds is not binary. More precisely it is a function that returns a value between 0 and 1 and increases with t.
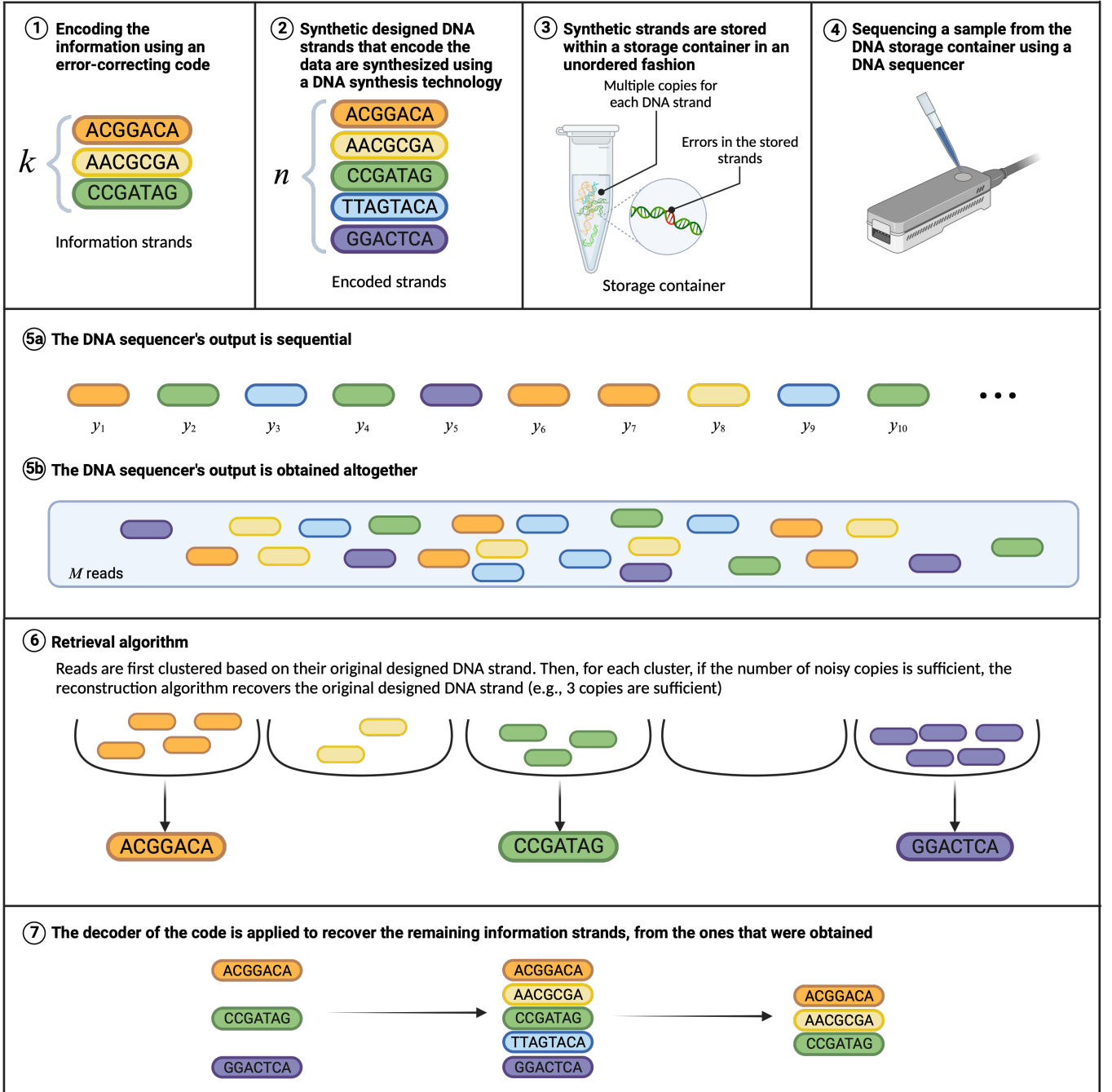
Fig. 1: The DNA storage pipeline.

III. THE COVERAGE DEPTH PROBLEM IN THE DNA STORAGE CHANNEL

In this section, we formally define the theoretical problems studied in this paper and review relevant existing results in the literature. We then present our contributions and the corresponding results for each problem.

A. Problems Definition

This work studies the required sample size to retrieve the information vector $\mathbf{U}$, or a specific subset of its symbols, 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.

According to our model characterization, we let $\nu_t^{\mathbf{p}}(\mathcal{C})$ be the random variable that governs the number of reads that should be sampled for successful decoding of $\mathbf{U}$. When $\mathcal{C}$ is an $[n, k]$ MDS code, this notation is replaced by $\nu_t^{\mathbf{p}}(n, k)$. The uniform distribution is denoted by $\mathbf{p}_u \triangleq (\frac{1}{n}, \ldots, \frac{1}{n})$ and

for brevity, we let $\nu_t(\mathcal{C}) \triangleq \nu_t^{\boldsymbol{p}_u}(\mathcal{C})$ and $\nu_t(n,k) \triangleq \nu_t^{\boldsymbol{p}_u}(n,k)$. The first two problems, which focus on retrieving the entire information vector $\mathbf{U}$, are defined below.

Problem 1. (The MDS coverage depth problem.) For given values of k and n, and a channel distribution $\boldsymbol{p}$ find the expectation and the probability distribution of the random variable $\nu_t^{\boldsymbol{p}}(n,k)$. That is, find the values of $\mathbb{E}\left[\nu_t^{\boldsymbol{p}}(n,k)\right]$ and $P[\nu_t^{\boldsymbol{p}}(n,k) > m]$ for any $m \in \mathbb{N}$.

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 $\mathcal{C}$ with dimension k and channel distributions $\boldsymbol{p}$. That is, find the value $\mathsf{M}^{opt}(k) \triangleq \liminf_{\mathcal{C}, \boldsymbol{p}} \{\mathbb{E}\left[\nu_t^{\boldsymbol{p}}(\mathcal{C})\right]\}$.

The third problem is related to the other setup, in which the user wishes to retrieve a subset of the k information strands (i.e., a subset of $\mathbf{U}$'s symbols). This subset can be described by an index set $I \subseteq [k]$, such that the set of information strands to be retrieved is $\mathbf{U}_I = \{\boldsymbol{u}_i : i \in I\}$. 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 governs the number of samples to recover the i-th information strand assuming noiseless channel with uniform distribution. 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.,

$$T_{\max}^{\mathcal{C}} \triangleq \max_{1 \le i \le k} \mathbb{E}[\tau_i(\mathcal{C})].$$

3) The average expected number of trials to retrieve an information strand, i.e.,

$$T_{\text{avg}}^{\mathcal{C}} \triangleq \frac{1}{k} \sum_{i=1}^{k} \mathbb{E}[\tau_i(\mathcal{C})].$$

When no coding is used, (i.e., $\mathbf{U} = \mathbf{X}$) $\mathcal{C}$ is removed from the notations.

B. 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 finding the expectation listed in Problem 1 is equivalent to the classical *coupon collector's problem* [15]. This problem was first studied by Feller [15] 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 of the coupons, the expected number of draws (i.e., sample size) required to get a single copy for each

coupon is $\mathbb{E}[\nu_1(n, k = n)] = nH_n = n \log n + \gamma n + \mathcal{O}(1)$, where H_n is the n-th harmonic number and $\gamma \sim 0.577$ is the Euler–Mascheroni constant. Furthermore, it was also proven [16] that $\mathbb{E}[\nu_1(n, k)] = n(H_n - H_{n-k})$. It is well-known that when $\lim_{n \to \infty} n - k = \infty$,[3] the expectation can be approximated by $\mathbb{E}[\nu_1(n, k)] \approx n \log(n) - n \log(n - k) = n \log(\frac{n}{n-k})$.

For noisy channels, i.e., $t > 1$, the problem is closely related to the classical *urn problem* [12], [25]. 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 $1 \le j \le n$, and we let $\boldsymbol{p} = (p_1, \dots, p_n)$. In [25], it was shown that in order to have t balls in each urn (or equivalently t copies per coupon), the expected sample size is $\mathbb{E}[\nu_t(n, k = n)] = n \log n + n(t-1) \log \log n + nC_t + o(n)$, where C_t is a constant that depends on t. Following that, Erdős and Rényi [12] proved that the distribution of this random variable is tightly concentrated around the expectation. More specifically, when drawing $n \log n + n(t-1) \log \log n + nx$ times, the probability to have at least t copies for n coupons is asymptotically equal to $e^{-\frac{e^{-x}}{(t-1)!}}$ for n large enough. Flajolet et al. [16] 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

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

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 the expectation value listed in Problem 1, not only for $\boldsymbol{p}_u$ but for any channel distribution. As can be seen, for practical purposes, the expression in (1) and its asymptotic behavior are not easy to calculate or to work with. Hence, in Section IV-B we solve a closely related problem and present a closed-form expression. Moreover, to the best of our knowledge, the other part of Problem 1, i.e., studying the cumulative probability distribution $P[\nu_t^{\boldsymbol{p}}(n,k) > m]$, is still open.

Another related problem was presented in [7] by Chandak et al. In their paper, the authors defined the *writing cost* as the number of synthesized bases per information bit, and the *reading cost* as the number of bases that have to be sequenced per information bit in order to retrieve the stored information. Their paper studies the tradeoffs between these two costs. They first showed that for the noiseless channels, the event of obtaining zero copies of a specific strand is equivalent to an erasure of this strand, which can be approximated as a Poisson random variable. Thus, they were able to compute the capacity of this channel and by this obtaining the tradeoffs of the costs. For the noisy channel, the authors suggested an LDPC-based scheme to improve the ratio between the costs, for more details see [7]. A related approach was discussed in [27], where the authors used an LDPC-based scheme for

---

[3]In this case, there exists $0 < a < 1$, such that for n large enough $k < an$.

error correction, together with retrieval algorithms, and showed experimentally that their approach reduces the reading costs compared to previously published works. In each of the works [7], [13], and [27], a specific practical coding solution was proposed and analyzed by the authors, demonstrating that they effectively reduce the ratio between writing and reading costs by using their selected codes. This work, however, provides a more general framework and defines the theoretical problems that should be studied to rigorously analyze the required coverage depth, focusing on understanding how different coding schemes affect the required number of reads[4]. This work initiates the theoretical study of these problems, which are related but distinct from the problems discussed in [7], [27]. It establishes theoretical bounds and explores optimal codes, including for random-access setups. Thus, the results presented in this paper are independent and complementary to any of the previously suggested solutions.

The problem of random access in DNA storage has already been addressed in several works; see e.g. [3], [22], [26], [39], [42]. The main goal is to support random access to specific DNA strands in the storage and this can be supported by the use of different primers for the different strands or physically storing strands in different storage containers. However, these solutions incur high costs, and thus the problem of storing strands together using these primers is still important and this work addresses it from a coding theory perspective.

C. Main Contributions

In this paper, we define a new family of problems that should be considered when designing DNA storage systems. Additionally, this work provides an extensive analysis and present results that enhance our understanding of the interplay between error-correcting codes, retrieval algorithms, and coverage depth. The main results with respect to each of the three problems we defined are listed below.

The MDS coverage depth problem (Problem 1) For this problem, we have the following results in Section IV.

1) We show in Theorem 1 that the value of $\mathbb{E}[\nu_1^P(n,k)]$ is minimized if and only if the channel has the uniform distribution.

2) We show in Theorem 3 and in Theorem 4 two upper bounds on the probability distribution of $P[\nu_t(n,k) > m]$. We further prove in Lemma 1 a lower bound on the probability $P[\nu_t(n,k) \leq m]$. Combining these results in Theorem 5 we prove that for any $\varepsilon > 0$,

$$\log\left(\frac{1}{1-R}\right) + f_c(n,R) \leq \mathbb{E}\left[\frac{\nu_t(n,k)}{n}\right]$$
$$\leq \left(\log\left(\frac{1}{1-R}\right) + t\log\log n + 2\log(t+1)\right) \cdot (1+2\varepsilon),$$

where $f_c(n,R) = \mathcal{O}(\frac{1}{n^2})$.

3) For practical purposes of DNA storage systems, it is sometimes required to plan ahead and set the number

of reads that should be sampled to guarantee successful decoding. Hence, we show in Theorem 6, that when sampling more than $r_E(n,k,t)$ reads, the expected number of encoded strands that cannot be recovered (i.e., have less than t copies) is at most $n-k$, which indicates on the probability of successful decoding. The value of $r_E(n,k,t)$ can be found in equation (10).

The coding coverage depth problem (Problem 2) We fully solve Problem 2 for the noiseless channel with uniform distribution. We show that MDS codes are optimal with respect to minimizing $\mathbb{E}[\nu_t(n,k)]$. We also show that for a fixed k, the larger n is, the smaller the value of $\mathbb{E}[\nu_t(n,k)]$ is. The results of this problem are given Section V.

The singleton coverage depth problem (Problem 3) We extensively study the singleton coverage depth problem for the case in which the channel is noiseless in Section VI. Our main results are summarized below.

1) We first study and fully solve the case in which $n = k$. In particular, we prove that if $n = k$ then the expected time to retrieve a singleton is minimized when no coding is used and it is equal to k and $T_{\max}^{\mathcal{C}} = T_{\text{avg}}^{\mathcal{C}} = k$ (see Lemma 2 and Claim 5).

2) Next, to study more involved cases, we define *retrieval sets* and *minimal retrieval sets*, which correspond to the (minimal) sets of encoded strands from which a specific target singleton information strand can be recovered. Using this property of codes, we analyze the expected time to retrieve a singleton given that its minimal retrieval sets are disjoint. See these results in Theorem 8 and Corollary 4. Moreover, in Corollary 3 we use Theorem 8 to conclude that the expected time to retrieve a singleton given that $\mathcal{C}$ is the simple parity $[k+1,k]$ code is k.

3) We extend the result of Corollary 3 to any systematic $[n,k]$ MDS code, by the construction and detailed evaluation of the corresponding generating function. That is, we show in Theorem 9, that for any $[n,k]$ MDS code $\mathcal{C}$ and any $i \in [k]$, $\mathbb{E}[\tau_i(\mathcal{C})] = T_{\max}^{\mathcal{C}} = T_{\text{avg}}^{\mathcal{C}} = k$.

4) We give two explicit code constructions (Construction 1 and Construction 2) for codes with k information strands for which $T_{\max}^{\mathcal{C}} < k$, i.e., $\mathbb{E}[\tau_i(\mathcal{C})] < k$ for all $i \in [k]$. Furthermore, we analyze the behavior of these codes both analytically and by computer simulations.

5) To conclude the analysis of the singleton coverage depth problem, we provide in Lemma 3 and in Theorem 13 two lower bounds on the value of $T_{\max}^{\mathcal{C}}$. Moreover, in Corollary 7, for n large enough, we show that for any (n,k) code $\mathcal{C}$, such that $R = \frac{k}{n}$, we have that $T_{\max}^{\mathcal{C}} \geq k\left(\frac{1}{R} + \frac{1-R}{R^2} \cdot \ln(1-R)\right)$. In particular, the latter implies that when R approaches zero, $T_{\max}^{\mathcal{C}} \geq \frac{k}{2}$, and when R approaches one, the lower bound approaches k from below.

---

[4]The key difference between the problems discussed in this paper and the problems discussed in [7], [27] is that while in [7], [27], the goal is to optimize the ratio between the writing cost and reading cost, in this work, the optimization is done on the number of reads, which is highly related to the reading cost.

IV. THE MDS COVERAGE DEPTH PROBLEM

In this section, we focus on the case discussed in Problem 1. That is, the case where the code $\mathcal{C}$ is is an MDS code. We start with the noiseless channel which refers to $t = 1$ in Section IV-A and the noisy channel is considered in Section IV-B.

A. The MDS Coverage Depth Problem - Noiseless Channel

In this section, we focus on the setup where $t = 1$. Hence, we can assume that the retrieval algorithm simply returns the sampled sequences and thus if $\boldsymbol{x}_i$ has at least one copy, i.e., $t \geq 1$, it is enough to retrieve it. Under this setup, the minimum sample size M is equivalent to the quantity which is governed by the random variable $\nu_1^{\boldsymbol{p}}(n, k)$. Using our notation, note that the expected value of $\nu_1^{\boldsymbol{P}}(n, k)$ is given in (2). In this case the distribution probability function for $\boldsymbol{p} = \boldsymbol{p}_u$ was studied in [12]. Clearly, when $k = 1$, we have that $\mathbb{E}[\nu_1(n, 1)] = 1$. Hence, this section is focused on the case where $k \geq 2$. Our main result is to show that $\mathbb{E}[\nu_1^{\boldsymbol{p}}(n, k)]$ is minimized when $\boldsymbol{p}$ is the uniform distribution and is bounded from below by $k \log e$ if $\frac{k}{n} = \Theta(1)$.

In light of the existing results, we first show that for the uniform channel distribution, when k is fixed, $\mathbb{E}[\nu_1(n, k)]$ decreases as n increases.

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

Proof: The proof follows by showing that $\mathbb{E}[\nu_1(n, k)]$ is a monotonic function that decreases with n. From [16] for any $n \in \mathbb{N}$ we have that

$$\mathbb{E}[\nu_1(n, k)] - \mathbb{E}[\nu_1(n + 1, k)] = \sum_{i=0}^{k-1} \frac{n}{n - i} - \sum_{i=0}^{k-1} \frac{n+1}{n+1-i}$$
$$= \sum_{i=0}^{k-1} \left(\frac{n}{n-i} - \frac{n+1}{n+1-i} \right)$$
$$= \sum_{i=0}^{k-1} \frac{i}{(n-i)(n+1-i)} > 0,$$

which completes the proof. ∎

In the next theorem it is shown that for MDS codes, $\mathbb{E}[\nu_1^{\boldsymbol{P}}(n, k)]$ is minimized when $\boldsymbol{p} = \boldsymbol{p}_u$.

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

Proof: By (1), which was proven originally in [16], we have that

$$\mathbb{E}[\nu_1^{\boldsymbol{P}}(n, k)] = \sum_{q=0}^{k-1} \int_0^\infty [u^q] \prod_{i=1}^n \left(1 + u(e^{p_i v} - 1)\right) e^{-v} dv$$
$$= \sum_{q=0}^{k-1} \int_0^\infty e^{-nv} \left(\sum_{\substack{I \subseteq [n] \\ |I|=q}} \prod_{i \in I} (e^{p_i v} - 1) \right) dv$$
$$= \int_0^\infty e^{-nv} \cdot \sum_{q=0}^{k-1} \left(\sum_{\substack{I \subseteq [n] \\ |I|=q}} \prod_{i \in I} (e^{p_i v} - 1) \right) dv. \quad (2)$$

Define $f(p_1, \ldots, p_n) \triangleq \sum_{q=0}^{k-1} \left(\sum_{\substack{I \subseteq [n] \\ |I|=q}} \prod_{i \in I} (e^{p_i v} - 1) \right)$. We show next that f is minimized if and only if $p_i = \frac{1}{n}$ for all $1 \leq i \leq n$. Furthermore, since f is minimized if and only if $\mathbb{E}[\nu_1^{\boldsymbol{P}}(n, k)]$ is minimized, this concludes the proof.

Define $g(\boldsymbol{p}) = -1 + \sum_{i=1}^n p_i$. Using Lagrange multipliers, the Lagrangian function is

$$\mathcal{L}(\boldsymbol{p}, \lambda) = f(\boldsymbol{p}) + \lambda g(\boldsymbol{p})$$
$$= \sum_{q=0}^{k-1} \left(\sum_{\substack{I \subseteq [n] \\ |I|=q}} \prod_{i \in I} (e^{p_i v} - 1) \right) - \lambda + \lambda \sum_{i=1}^n p_i.$$

We are looking for values of $\boldsymbol{p}$, that satisfy

$$\frac{\partial \mathcal{L}(\boldsymbol{p}, \lambda)}{\partial \lambda} = -1 + \sum_{i=1}^n p_i = 0, \quad (3)$$

and for all $1 \leq i \leq n$,

$$\frac{\partial \mathcal{L}(\boldsymbol{p}, \lambda)}{\partial p_i} = \lambda + \sum_{q=1}^{k-1} \left(\sum_{\substack{I \subseteq [n] \setminus \{i\} \\ |I|=q-1}} v e^{p_i v} \prod_{j \in I} (e^{p_j v} - 1) \right) = 0,$$

which is equivalent to

$$\lambda = -v e^{p_i v} \sum_{\substack{I \subseteq [n] \setminus \{i\} \\ |I| < k-1}} \prod_{j \in I} (e^{p_j v} - 1). \quad (4)$$

Hence, for any $1 \leq i < i' \leq n$ we have that

$$e^{p_i v} \sum_{\substack{I \subseteq [n] \setminus \{i\} \\ |I| < k-1}} \prod_{j \in I} (e^{p_j v} - 1) = e^{p_{i'} v} \sum_{\substack{I \subseteq [n] \setminus \{i'\} \\ |I| < k-1}} \prod_{j \in I} (e^{p_j v} - 1).$$

By reorganizing the latter equation, we have that for any $1 \leq i < i' \leq n$,

$$(e^{p_i v} - e^{p_{i'} v}) \sum_{\substack{I \subseteq [n] \setminus \{i, i'\} \\ |I| < k-1}} \prod_{j \in I} (e^{p_j v} - 1)$$
$$= (e^{p_i v} - e^{p_{i'} v}) \sum_{\substack{I \subseteq [n] \setminus \{i, i'\} \\ |I| < k-2}} \prod_{j \in I} (e^{p_j v} - 1),$$

which is equivalent to

$$(e^{p_i v} - e^{p_{i'} v}) \sum_{\substack{I \subseteq [n] \setminus \{i, i'\} \\ |I| = k-2}} \prod_{j \in I} (e^{p_j v} - 1) = 0.$$

Hence, we have that $p_i = p_{i'}$ or that $|\{j : j \neq i, i', p_j > 0\}| < k - 2$. To complete the proof, let us show that the minimum is not attained for any $\boldsymbol{p}$ such that, $|\mathsf{supp}(\boldsymbol{p})| < n$. We prove the latter using an induction on n. For clarity, we will use the notation f_n to indicate the relevant value of n and $\boldsymbol{p}_{\boldsymbol{u}}^n \triangleq (\frac{1}{n}, \ldots, \frac{1}{n})$. The base case in which $n = 2$ can be verified manually. This implies that $\boldsymbol{p} = \boldsymbol{p}_{\boldsymbol{u}}^2 = (\frac{1}{2}, \frac{1}{2})$ is the only minimum point for f_2. Assume the claim holds up to n, and let us prove its correctness for $n+1$. Let $\boldsymbol{p} = (p_1, \ldots, p_{n+1})$ be a minimum point for f_{n+1}, and assume by contradiction that $|\mathsf{supp}(\boldsymbol{p})| < n + 1$ and further assume w.l.o.g. that $p_{n+1} = 0$. Define $\boldsymbol{p}' = (p_1, \ldots, p_n)$ and note that $f_{n+1}(\boldsymbol{p}) = f_n(\boldsymbol{p}')$. By the induction assumption, we know that a minimum point of f_n has a support of size n and hence, by the analysis of the Lagrangian function, we have that f_n has a unique minimum point at $\boldsymbol{p}_{\boldsymbol{u}}^n$. Therefore, we have that

$$f_{n+1}(\boldsymbol{p}) = f_n(\boldsymbol{p}') \geq f_n(\boldsymbol{p}_{\boldsymbol{u}}^n),$$

and equality is obtained if and only if $\boldsymbol{p}' = \boldsymbol{p_u^n}$. Moreover, Claim 1 implies that $f_n(\boldsymbol{p_u^n}) > f_{n+1}(\boldsymbol{p_u^{n+1}})$, and thus,

$$f_{n+1}(\boldsymbol{p}) \geq f_n(\boldsymbol{p_u^n}) > f_{n+1}(\boldsymbol{p_u^{n+1}}),$$

which is a contradiction. Thus, we get that $|\mathsf{supp}(\boldsymbol{p})| = n+1$, which implies that the only minimum point of f_{n+1} is $\boldsymbol{p_u^{n+1}}$. ∎

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

Corollary 1. For any channel distribution $\boldsymbol{p}$ it holds that,

$$\mathbb{E}[\nu_1^{\boldsymbol{p}}(n,k)] \geq \mathbb{E}[\nu_1(n,k)] = \sum_{i=0}^{k-1} \frac{n}{n-i}, \qquad (5)$$

and if $\lim_{n\to\infty} n - k = \infty$ then $\sum_{i=0}^{k-1} \frac{n}{n-i} \approx n\log(\frac{n}{n-k})$. Moreover (11) holds with equality if and only if $\boldsymbol{p} = \boldsymbol{p}_u$.

Finally, we give the asymptotic value for the minimum expected sample size for the noiseless channel, $\mathbb{E}[\nu_1(n,k)]$, where the proof can be found in Appendix A.

Theorem 2. Let R be a constant, $0 < R < 1$. Then, we have that

$$\lim_{n\to\infty} \frac{\mathbb{E}[\nu_1(n,k=\lfloor nR \rfloor)]}{k} = \frac{1}{R}\log\left(\frac{1}{1-R}\right).$$

Furthermore, consider a sequence of MDS codes $\{\mathcal{C}_i\}_{i=1}^{\infty}$ with parameters $[n_i, k_i]$ such that $\lim_{i\to\infty} k_i/n_i = 0$. Then,

$$\lim_{i\to\infty} \frac{\mathbb{E}[\nu_1(n_i,k_i)]}{k_i} = 1.$$

B. The MDS Coverage Depth Problem - The Noisy Channel

The main goal of this section is to address Problem 1 for the noisy channel under the uniform distribution. 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 some $t > 1$ reads, which are noisy copies of it, and cannot be retrieved given less than t reads. Similarly to the previous section, it is enough to successfully decode k (or more) sequences $\boldsymbol{x}_i$ in order to retrieve the stored information and so under this setup the minimum sample size for our problem is equivalent to the quantity $\nu_t(n,k)$ where $t > 1$.

It should be noted that as listed in the related work section, the first part of Problem 1, i.e., the value of $\mathbb{E}[\nu_t(n,k)]$ is known [16] and is given in (1). However, it is not a closed-form expression, and in this section, we give several closed-form expressions that bound this value and thus extend the known result. Furthermore, the most related result regarding the probability distribution $P[\nu_t(n,k) > m]$ was given in [12]. The authors showed that for $n = k$ any $x \in \mathbb{R}$, the probability satisfies

$$P[\nu_t(n,n) > n\log n + (t-1)\log\log n + nx] \leq e^{-\frac{e^{-x}}{(t-1)!}}.$$

In this section, we extend the latter result, by providing several bounds for the case when $k < n$, which is assumed for the rest of this section. Our main results for this case are stated in Theorem 3 and in Lemma 1. To discuss these results,

we first define the following value. Given n, k, and t as stated above, we define

$$r(n,k,t) \triangleq n\log\left(\frac{n}{n-k}\right) + nt\log\log n + 2n\log(t+1). \quad (6)$$

In Theorem 3, it is shown that when n is large enough, the probability that more than $r(n,k,t)$ reads are required to retrieve the information i.e., $P[\nu_t(n,k) > r(n,k,t)]$ approaches zero.

Furthermore, for the case in which $k = Rn$, where $0 < R < 1$ is a fixed constant, the value of $r(n,k,t)$ can be reduced by replacing the expression $\log\log(n)$ with any function of n that approaches to infinity with n. To conclude this discussion, we also show in Lemma 1 that for any c, the probability that less than $n\log(\frac{n}{n-k}) - nc$ reads are enough to retrieve the information is bounded from above by $e^{-c}(1+\frac{1}{n-k})$. We start by showing that for any $\varepsilon > 0$, $P[\nu_t(n,k) \leq r(n,k,t)] \geq 1-\varepsilon$ for n large enough.

Theorem 3. For any $\varepsilon > 0$ and any integer n, such that $n > e^{\frac{6t\cdot2^{t-1}}{\varepsilon}} \geq 16$, we have that,

$$P[\nu_t(n,k) \leq r(n,k,t)] \geq 1 - \varepsilon.$$

Proof: To prove the statement in the theorem it is sufficient to show that

$$P[\nu_t(n,k) > r(n,k,t)] < \varepsilon.$$

Denote $r \triangleq r(n,k,t)$ and recall that within the context of the urn problem (see Section III-B), the random variable $\nu_t(n,k)$ denotes the number of balls (or rounds) until we have a set of k urns where each urn has at least t balls. Hence, we show that if the number of balls thrown is at least r, then the probability of having $n - k + 1$ or more urns which are *not* filled with t balls is approaching zero. The approach leveraged in the proof is inspired by a technique first employed by Erdős and Rényi in [12]. Let us define the following event.

$E_t^{(r)}$: After r rounds, there exists a set S_t, of $n - k + 1$ urns, each containing less than t balls.

Next, we show that the probability of $E_t^{(r)}$ approaches zero when n is large. To this end, we define $z_i(n,r)$ for $1 \leq i \leq n$, as a random variable that governs the number of balls in the i-th urn, after r draws. For n large enough, the probability that urn i has at most $t-1$ balls after r draws is denoted by $P[z_i(n,r) \leq t-1]$ and is given by,

$$P[z_i(n,r) \leq t-1] = \sum_{j=0}^{t-1} \binom{r}{j}\left(\frac{1}{n}\right)^j\left(1-\frac{1}{n}\right)^{r-j}$$

$$\leq t\cdot\binom{r}{t-1}\left(\frac{1}{n}\right)^{t-1}\left(1-\frac{1}{n}\right)^{r-(t-1)}$$

$$\leq t\cdot\left(\frac{r\cdot e}{t-1}\right)^{t-1}\left(\frac{1}{n}\right)^{t-1}\left(1-\frac{1}{n}\right)^{r-(t-1)},$$

where the first inequality is proven in Claim 6 in Appendix B, and the last inequality follows from the fact that $\binom{r}{t-1} \leq (\frac{re}{t-1})^{t-1}$. Note that $(\frac{e}{t-1})^{t-1} < 3$, for $t > 1$. Thus,

$$P[z_i(n,r) \leq t-1] \leq 3t\cdot\left(\frac{r}{n}\right)^{t-1}\left(1-\frac{1}{n}\right)^{n\left(\frac{r}{n}-\frac{t-1}{n}\right)}.$$

We have that,

$$P\left[z_i(n,r) \leq t-1\right]$$

$$\leq 3t\left(\log\left(\frac{n}{n-k}\right) + t\log\log(n) + 2\log(t+1)\right)^{t-1} e^{\frac{-r+t-1}{n}}$$

$$\leq 3t(2\log n)^{t-1}\left(\frac{n-k}{n}\right)\left(\frac{1}{\log^t n}\right)\left(\frac{1}{(t+1)^2}\right)e^{\frac{t-1}{n}}$$

$$= 3t \cdot \frac{e^{\frac{t-1}{n}}}{(t+1)^2} \cdot \frac{(2\log n)^{t-1}}{\log^t(n)} \cdot \frac{n-k}{n}$$

$$= 3t \cdot \frac{e^{\frac{t-1}{n}}}{(t+1)^2} \cdot \frac{2^{t-1}}{\log(n)} \cdot \frac{n-k}{n},$$

where the second inequality holds since for n large enough $\log(\frac{n}{n-k}) + t\log\log(n) + 2\log(t+1) \leq (2\log n)$. It should be noted that for $n > t$, which is the case of our interests, we have that $3t \cdot \frac{e^{\frac{t-1}{n}}}{(t+1)^2} \leq 6t$, and hence,

$$P\left[z_i(n,r) \leq t-1\right] \leq 6t \cdot \frac{2^{t-1}}{\log(n)} \cdot \frac{n-k}{n}.$$

Now let us define a random variable Y as the number of urns with less than t balls. From the linearity of expectation, regardless if the urns are independent or not, the expected number of urns that have less than t balls is,

$$\mathbb{E}[Y] = \Sigma_{i=1}^n \mathbb{E}[z_i(n,r)]$$

$$= n \cdot P\left[z_i(n,r) \leq t-1\right] \leq (n-k) \cdot 6t \cdot \frac{2^{t-1}}{\log(n)},$$

where the last inequality holds for n large enough.

Note that

$$P\left[E_t^{(r)}\right] = P[Y \geq n-k+1],$$

and hence by Markov's inequality, we can conclude that,

$$P[Y \geq n-k+1] \leq \frac{\mathbb{E}[Y]}{n-k+1} < 6t \cdot \frac{2^{t-1}}{\log(n)}.$$

Thus, we get that $P[E_t^{(r)}] \to 0$ for n large enough which implies the statement in the theorem. ∎

For fixed-rate codes, i.e., for the case where $k = Rn$, when $0 < R < 1$, and R is a fixed constant (when n grows), we present a stronger result in the next theorem. The proof of this theorem can be found in Appendix B.

Theorem 4. Let $f : \mathbb{N} \to \mathbb{R}$ be a function such that $\lim_{n\to\infty} f(n) = \infty$, and let

$$r_f(n, k=Rn, t) \triangleq n\log\left(\frac{1}{1-R}\right) + ntf(n) + 2n(t+1). \quad (7)$$

Then, for n large enough, it holds that

$$P\left[\nu_t(n,k) > r_f(n,k,t)\right] \leq 6t^t \frac{(2 \cdot f(n))^{t-1}}{e^{t \cdot f(n)}} \cdot (1-R).$$

Theorem 4 draws a connection between the sample size and the probability of successful retrieval when using fixed-rate codes. In particular, using the results of Theorem 4 one can pick any function $f(n)$ that approaches infinity as slowly (or fast) as possible to get an upper bound on this probability which gets bigger (or smaller).

In the next lemma, an upper bound on the probability $P[\nu_t(n,k) \leq n\log\left(\frac{n}{n-k}\right) - nc]$, for any positive $c \in \mathbb{R}$, is given. The proof is based on the fact that $\nu_t(n,k) \geq \nu_1(n,k)$ and can be found in Appendix C.

Lemma 1. For any $c > 0$, and any $t \geq 1$ it holds that,

$$P\left[\nu_t(n,k) \leq n\log(\frac{n}{n-k}) - nc\right] \leq e^{-c}\left(1 + \frac{1}{n-k}\right).$$

Combining Lemma 1 and Theorem 3, and assuming t is a constant with respect to n, in the next theorem we show upper and lower bounds on $\mathbb{E}\left[\frac{\nu_t(n,k)}{n}\right]$, and the detailed proof can be found in Appendix D.

Theorem 5. For any $\varepsilon > 0$, there exists n_ε, such that for any $n > n_\varepsilon$ we have that,

$$\log\left(\frac{1}{1-R}\right) + f_c(n,R) \leq \mathbb{E}\left[\frac{\nu_t(n,k)}{n}\right]$$

$$\leq \left(\log\left(\frac{1}{1-R}\right) + t\log\log n + 2\log(t+1)\right) \cdot (1+2\varepsilon),$$

where

$$f_c(n,R) = \frac{1}{2n}\left(1 - \frac{1}{1-R}\right) - \sum_{h=1}^\infty \frac{B_{2h}}{2hn^{2h}}\left(1 - \frac{1}{(1-R)^{2h}}\right)$$

$$= \mathcal{O}\left(\frac{1}{n^2}\right),$$

and B_h denotes the h-th Bernoulli number.

For practical purposes of DNA storage systems, it is sometimes required to plan ahead and sample the number of reads that guarantees successful decoding with high probability. Hence, we turn to the following strongly related problem and give a closed-form expression to the corresponding value. Turning back to the urn problem terminology, we define $X^{(r)}$ as the number of urns that are not filled with at least t balls after r rounds. 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 balls is at most $n-k$. That is, to find r_E, such that for any $r \geq r_E$, we have that $\mathbb{E}[X^{(r)}] \leq n-k$. In order to derive this result, we first consider the probability that any fixed urn is *not filled* with t or more balls by the r-th round. This probability is given by,

$$p = \sum_{j=0}^{t-1}\binom{r}{j}n^{-j}\left(1 - \frac{1}{n}\right)^{r-j} \leq e^{-rD(\frac{t-1}{r}||\frac{1}{n})},$$

where the last inequality follows from Chernoff bound [9] for $r \geq n(t-1)$, and $D(a||p)$ is the Kullback–Leibler divergence [10] which is given by

$$D(a||p) \triangleq a\log_2\frac{a}{p} + (1-a)\log_2\frac{1-a}{1-p}.$$

Under our 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 $1 \leq i \leq n$. Note that $X^{(r)} = \sum_{i=1}^n X_i^{(r)}$ is the number of urns that are not filled with at least t balls after r rounds, which implies that the number of urns that have at least t 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,

$$\mathbb{E}[X^{(r)}] = np$$
$$\leq ne^{-(t-1)\log_2\left(\frac{n(t-1)}{r}\right)-(r-(t-1))\log_2\left(\frac{(r-(t-1))n}{r(n-1)}\right)}. \quad (8)$$

The next claim will be used in the derivation to follow and its proof can be found in Appendix F.

Claim 2. For $r \geq n(t-1)$, we have that $\mathbb{E}[X^{(r)}] \leq n - k$, if,

$$-\frac{r}{n(t-1)}e^{-\frac{r}{n(t-1)}} \geq -\frac{1}{e}\left(1-\frac{k}{n}\right)^{\frac{\log 2}{t-1}}. \quad (9)$$

Using known results on the Lambert W function [11, Section IV], [8, Theorem 1], the values of r for which (9) holds can be concluded. This is summarized in the next theorem, and the complete proof can be found in Appendix F. For $0 < R < 1$, we denote,

$$r_E(n, k = Rn, t) \triangleq$$
$$n(t-1)-n\log 2\log(1-R)+n(t-1)\sqrt{-\frac{2\log 2}{t-1}\log(1-R)}. \quad (10)$$

Theorem 6. Let $R = \frac{k}{n}$. For any $r \geq r_E(n, k, t)$, we have that $\mathbb{E}[X^{(r)}] \leq n - k$.

At this point, we would like to shed some light on the relation between Theorem 3, Theorem 4 and Theorem 6. In our setup, which uses the urn problem terminology, it is assumed that r balls are thrown into n unique urns, and we are interested in the event that at least k of these urns contain at least t balls each. This scenario can be parameterized in two different ways; (a) the number of balls that need to be thrown, and (b) the number of urns that contain $t-1$ or less balls. The random variable $\nu_t(n, k)$ governs the value in (a), assuming that the value in (b) is fixed. Analogously, the random variable $X^{(r)}$ governs the value in (b), assuming that the value in (a) is fixed.

In the case where the probability distribution is tightly concentrated (i.e., where $\nu_t(n, k)$ is tightly concentrated around its mean and similarly for $X^{(r)}$), one would expect these two quantities to coincide. Fig. 2, shows results from computer simulations we made to demonstrate the results of Theorem 3 and Theorem 6. In the presented simulation we used $n = 100,000$ urns, $R \in \{0.5, 0.8\}$, $k = Rn$, and $t = 5$. In each simulation r balls are drawn, each inserted into one of the urns randomly, and the simulation is considered as success if it ends with at least k urns, each with at least t balls. For any value of r, the presented result is the fraction of successful simulations out of $1,000$ simulations we have made per r. The Y-axis shows the fraction of successful experiments, and the X-axis shows the number of draws r normalized by $n\log(\frac{1}{1-R})$. It can be seen that the success rate of both values presented in Theorem 3 ($r(n, k, t)$) and Theorem 6 ($r_E(n, k, t)$) are 1.

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 [13], 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 is 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 [13] 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 fixed 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 in this case. We note that this setup is easier to analyze, and the following results can be derived using similar techniques as in the classical coupons collector's problem [15]; see Appendix G.

Theorem 7. For any $k \leq n$,

$$\mathbb{E}[\omega_\alpha(n, k)] = \frac{n}{\alpha}(H_n - H_{n-k}).$$

V. THE CODING COVERAGE DEPTH PROBLEM - NOISELESS CHANNEL

In this section we fully solve Problem 2 in the case $t = 1$ and show that MDS codes are optimal for any categorical channel distribution.

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 minimizes the expectation of $\nu_1^{\boldsymbol{p}}(\mathcal{C})$ compared to any other length-n codes. This can be verified by showing that the number of subsets of size k which are sufficient to retrieve the information is maximized when an MDS code is used.

Claim 3. Given k, n, and $\boldsymbol{p}$, assume that $\mathcal{C}$ is an (n, k) code. Then, it holds that, $\mathbb{E}[\nu_1^{\boldsymbol{p}}(n, k)] \leq \mathbb{E}[\nu_1^{\boldsymbol{p}}(\mathcal{C})]$, where equality is obtained if and only if $\mathcal{C}$ is an MDS code.

Proof: Given a sample of size M, we denote by $J \subseteq [n]$ the indices of the unique strands that are represented in this sample. If $|J| < k$ then it is impossible to successfully decode the information, which follows since the dimension of the code is k. Otherwise, when $|J| \geq k$, any $[n, k]$ MDS code can decode the stored information, while if $\mathcal{C}$ is not an MDS code there exists J' of size k from which the stored information cannot be decoded using $\mathcal{C}$. Therefore, if $\mathcal{C}$ is not an MDS code, for any $J \subseteq [n]$, if the information can be deocded using $\mathcal{C}$, then it can also be deocdedby an $[n, k]$ MDS code. This implies the inequality stated in the theorem, where equality holds if and only if $\mathcal{C}$ is an MDS code. ∎

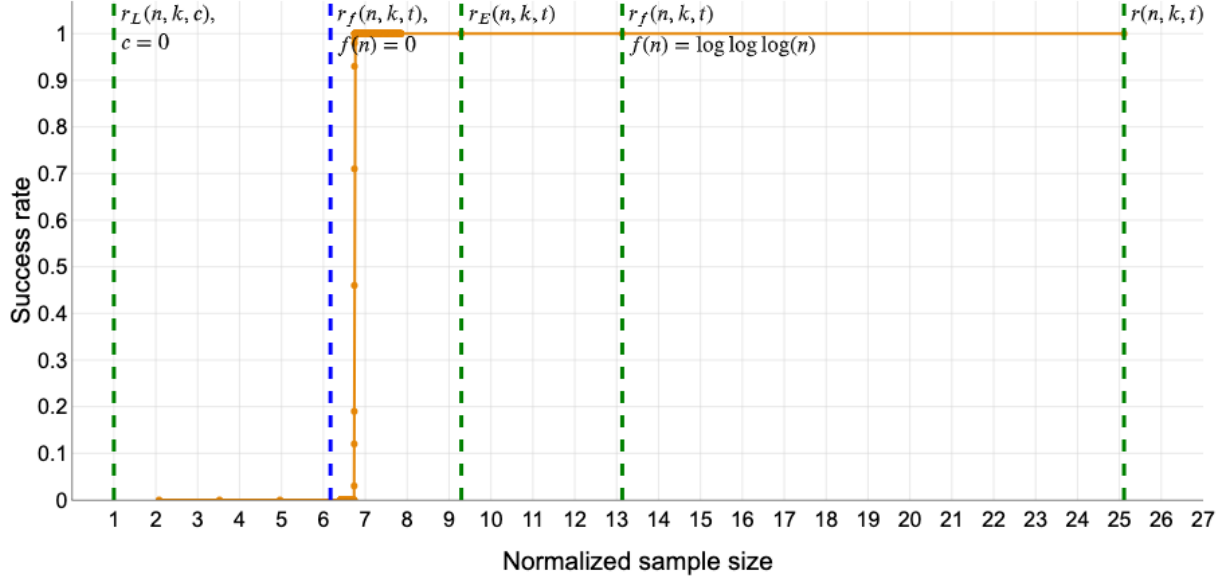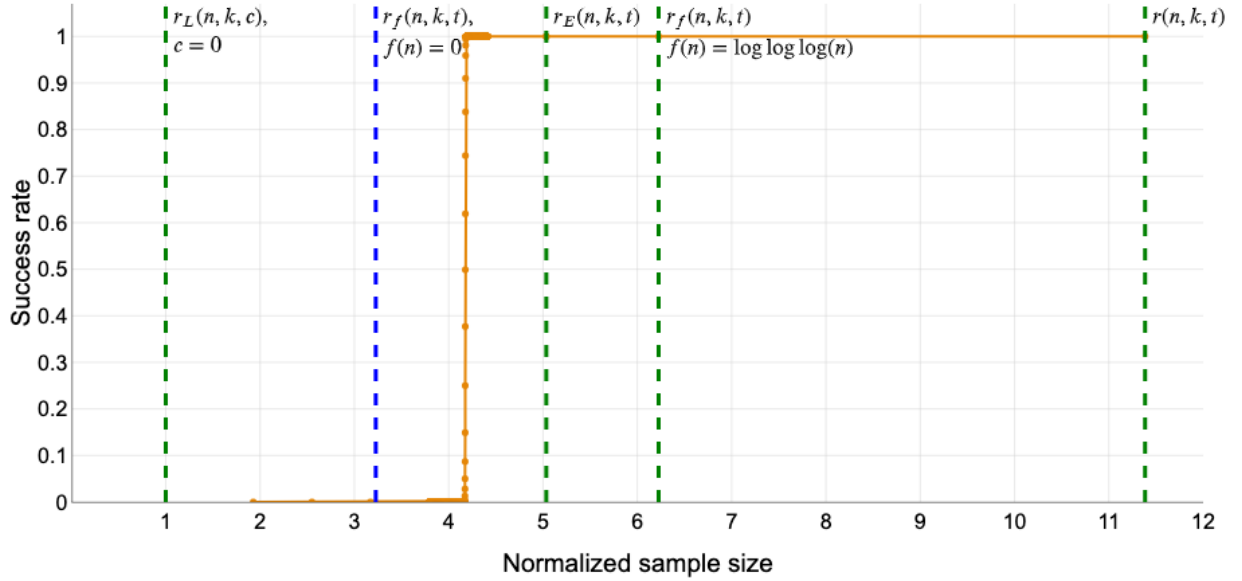Next, based on Corollary 1 and Claim 3, we obtain the next corollary

(a) $R = 0.5$

(b) $R = 0.8$

Fig. 2: Simulation results of the success rate (fraction of successful experiments) as a function of the number of draws. The X-axis shows the number of draws (normalized by $n \log(\frac{1}{1-R})$), while the Y-axis shows the fraction of simulations in which there were at least k urns with t balls each. The parameters used in the simulations were $n = 100,000$, $t = 5$, and for each number of draws, we had $1,000$ simulations. It can be seen that for both Theorem 3 and Theorem 6 the success rate of 1, as expected.

Corollary 2. For any channel distribution $\boldsymbol{p}$ and any (n,k) code $\mathcal{C}$, it holds that,

$$\mathbb{E}[\nu_1^{\boldsymbol{p}}(\mathcal{C})] \geq \mathbb{E}[\nu_1^{\boldsymbol{p}}(n,k)] \geq \mathbb{E}[\nu_1(n,k)] = \sum_{i=0}^{k-1} \frac{n}{n-i}, \quad (11)$$

and if $\lim_{n\to\infty} n - k = \infty$ then $\sum_{i=0}^{k-1} \frac{n}{n-i} \approx n\log(\frac{n}{n-k})$. Moreover (11) holds with equality if and only if $\boldsymbol{p} = \boldsymbol{p}_u$ and $\mathcal{C}$ is an MDS code.

Additionally, from Claim 1 we know that for fixed k, the larger n is, the smaller the value of $\mathbb{E}[\nu_t(n,k)]$ is and the value of $\lim_{n\to\infty} \frac{\mathbb{E}[\nu_1(n,k=\lfloor nR\rfloor)]}{k}$ is given in Theorem 2. Together, these results fully solves Problem 2.2 for the case $t = 1$. While this section fully solves the coverage depth problem for $t = 1$, the case where the channel is noisy (i.e. $t > 1$) remains a challenge that is left for future work.

VI. RANDOM ACCESS

In this section we study the problem of optimizing the sample size for random access queries in DNA storage systems. Recall that, in this problem, a vector of k information strands each of length ℓ, $\mathbf{U} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k) \in (\Sigma^\ell)^k$, is encoded into a vector of n strands, each of length ℓ, $\mathbf{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n) \in (\Sigma^\ell)^n$ that are stored in the DNA storage channel as described in Section II. Later, the user wishes to retrieve a single information strand $\boldsymbol{u}_i$ for some $i \in [k]$. Unless stated otherwise, we assume the channel is uniform and noiseless.

We start by studying the case when the number of information strands matches the number of coded strands (i.e. $n = k$), and prove that the optimal retrieval strategy involves no coding, resulting in an expected retrieval time of k. These preliminary results are given in Section VI-A. Next, in Section VI-B, we extend our insights to more involved cases, including systematic MDS codes, affirming that the expected retrieval time remains k for any information strand. Furthermore, in Section VI-C, we present explicit code constructions achieving expected retrieval times below k and evaluate their performance analytically and through simulations. Finally, Section VI-D concludes this analysis with lower bounds on the maximum expected retrieval time in different scenarios.

A. Preliminary Results

Recall that given an (n,k) code $\mathcal{C}$, for $i \in [k]$, we denote by $\tau_i(\mathcal{C})$ the random variable that governs the number of samples to recover the i-th information strand. The next lemma fully solves Problem 3 when no coding is used.

Lemma 2. Let $n \geq 1$. For any $1 \leq i \leq n$, we have that
  1) $\mathbb{E}[\tau_i] = n$ and $T_{\max} = T_{\text{avg}} = n$.
  2) For any $r \in \mathbb{N}$ we have that $P[\tau_i > r] = \left(1 - \frac{1}{n}\right)^r$, and $P[\tau_i = r] = \frac{1}{n} \cdot \left(1 - \frac{1}{n}\right)^{r-1}$.

Proof: As no coding is used, $\mathbf{U} = \mathbf{X}$ and hence for any i, τ_i governs the number of draws until we draw $\boldsymbol{x}_i = \boldsymbol{u}_i$ once. For the first part, note that for any i, τ_i has geometric

distribution with success probability $p = \frac{1}{n}$ and hence we have that $\mathbb{E}[\tau_i] = p^{-1} = n$ which implies that

$$T_{\max} = \max_{1 \leq i \leq n} \mathbb{E}[\tau_i] = p^{-1} = n,$$

and

$$T_{\text{avg}} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[\tau_i] = \frac{np^{-1}}{n} = p^{-1} = n.$$

For the second part we have that $\tau_i > r$ for an integer r only if $\boldsymbol{u}_i$ was not sampled in the first r trials, and hence

$$P[\tau_i > r] = (1-p)^r = \left(1 - \frac{1}{n}\right)^r,$$

and

$$P[\tau_i = r] = \frac{1}{n} \cdot \left(1 - \frac{1}{n}\right)^{r-1}.$$

$\blacksquare$

The proof follows the same ideas as the proof for the coupon collector's problem and is given in Appendix H for completeness.

Claim 4. For any (n,k) code $\mathcal{C}$ and any $J \subseteq [n]$ of size ρ we have that $\mathbb{E}[\widehat{\tau}_J(\mathcal{C})] = nH_\rho$.

For the rest of this section, it is assumed that $\mathcal{C}$ is an (n,k) code and $\mathbf{X}$ is the encoded codeword of the information vector $\mathbf{U}$. The structure of $\mathcal{C}$ defines for each information strand all the possible sets of encoded strands that are sufficient for its recovery. This concept is similar to recovery sets in *locally repairable codes* [28] as well as the ones with *availability* [14], [21], [37].

This can be defined formally as follows.

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 the inclusion relation).

Throughout the rest of the paper, we sometimes use $\mathcal{D}(i)$ with a slight abuse of notation, where the retrieval sets are expressed in terms of the encoded strands rather than their indices, to simplify the discussion.

We say that an (n,k) code $\mathcal{C}$ is a *systematic* code if for any $i \in [k]$ it holds that $\boldsymbol{u}_i$ has a retrieval set of size one. In other words, $\mathcal{C}$ is systematic if for any $i \in [k]$ we have that

$$\min\{|J| : J \in \mathcal{D}(i)\} = 1.$$

Next, we consider the case of non-systematic codes for $k = n$ (in particular $\mathbf{U} \neq \mathbf{X}$). Since $\mathbf{X}$ and $\mathbf{U}$ have the same length, given any set of strands $\{\boldsymbol{x}_i : i \in J\}$, we can recover at most $|J|$ information strands from $\mathbf{U}$. Our goal is to extend Lemma 2 to the coded case when $k = n$ using this basic insight.

Claim 5. For any code $(n = k, k)$ $\mathcal{C}$, we have that $T_{\max}^{\mathcal{C}} \geq T_{\max} = n$ and $T_{\text{avg}}^{\mathcal{C}} \geq T_{\text{avg}} = n$, where equality is obtained if and only if $\mathcal{C}$ is systematic. In particular, if we let ρ_i be the

size of the smallest retrieval set for the information strand $\boldsymbol{u}_i$, then

1) $\mathbb{E}[\tau_i(\mathcal{C})] = nH_{\rho_i}$,
2) $T_{\max}^{\mathcal{C}} = nH_\rho$, where $\rho \triangleq \max_i \rho_i$,
3) $T_{\text{avg}}^{\mathcal{C}} = \sum_{i=1}^{n} H_{\rho_i}$.

Proof: If each $\boldsymbol{u}_i$ can be retrieved from a single strand $\boldsymbol{x}_j$ (i.e., $\mathcal{C}$ is a systematic code), then similarly to the proof of Lemma 2 we have that $T_{\max}^{\mathcal{C}} = T_{\text{avg}}^{\mathcal{C}} = \mathbb{E}[\tau_i(\mathcal{C})] = n$, for any $i \in [n]$. Otherwise, assume w.l.o.g. that $\boldsymbol{u}_1$ cannot be retrieved from a single strand and let $J \subseteq [n]$ be a set of minimal size $|J| = \rho_1$ such that $J \in \mathcal{D}(1)$. By the latter observation and since it is possible to retrieve any information strand $\boldsymbol{u}_i$ from all the n strands, the fact that $n = k$ implies that if there exists $J' \subseteq [n]$, such that J' is a retrieval set of $\boldsymbol{u}_1$ (i.e., $J' \in \widehat{\mathcal{D}}(1)$) then J' contains J. Hence, $|\mathcal{D}(1)| = 1$, i.e., the set J is the only minimal retrieval set of $\boldsymbol{u}_1$, and by Claim 4, we have that $\mathbb{E}[\tau_i(\mathcal{C})] = \mathbb{E}[\widehat{\tau}_J(\mathcal{C})] = nH_{\rho_1} > n$, where the last inequality holds since $|J| = \rho_1 > 1$. Thus,

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

and

$$T_{\text{avg}}^{\mathcal{C}} = \frac{1}{k} \sum_{i=1}^{k} \mathbb{E}[\tau_i(\mathcal{C})] = \frac{1}{n} \sum_{i=1}^{n} nH_{\rho_i} = \sum_{i=1}^{n} H_{\rho_i}.$$

Note that $H_{\rho_i} \ge 1$ for any $i \in [k]$ and since $\rho_1 > 1$, we have that $H_{\rho_1} > 1$. Hence $T_{\max}^{\mathcal{C}} > n$ and $T_{\text{avg}}^{\mathcal{C}} > n$ which completes the proof. ∎

B. The Singleton Coverage Depth Problem

We continue by studying cases where $n > k$. Next, the case where the minimal retrieval sets are disjoint is considered. We start with the case in which a strand $\boldsymbol{x}_i$ has exactly two minimal retrieval sets $\mathcal{D}(i) = \{A, B\}$ and $A \cap B = \emptyset$, while the next example considers the simple parity code which is a special instance of this case.

Example 1. Let $\mathcal{C}$ be the $(4,3)$ parity code. We have that $\mathbf{X} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3, \boldsymbol{x}_4)$, where

$$\boldsymbol{x}_4 = \boldsymbol{u}_1 + \boldsymbol{u}_2 + \boldsymbol{u}_3.$$

Since the code is symmetric, let us consider w.l.o.g. $\boldsymbol{u}_1$. Note that $\mathcal{D}(1) = \{\{\boldsymbol{u}_1\}, \{\boldsymbol{u}_2, \boldsymbol{u}_3, \boldsymbol{x}_4\}\}$ and the two retrieval sets are disjoint. Hence, we cannot recover $\boldsymbol{u}_1$ from a series of r draws only if the series of draws does not contain $\boldsymbol{u}_1$, and it either contains one unique strand or two unique strands. Hence,

$$\mathbb{E}[\tau_1(\mathcal{C})] = \sum_{r=0}^{\infty} P[\tau_i(\mathcal{C}) > r]$$

$$= 1 + \sum_{r=1}^{\infty} \left(3 \cdot \frac{1}{4^r} + \binom{3}{2} \sum_{j=1}^{r-1} \binom{r}{j} \frac{1}{4^j} \cdot \frac{1}{4^{r-j}} \right)$$

$$= 1 + 3 \sum_{r=1}^{\infty} \frac{1}{4^r} + 3 \sum_{r=1}^{\infty} \frac{2^r - 2}{4^r}$$

$$= 1 + 3 \cdot \frac{1}{3} + 3 \cdot \frac{1}{3} = 3.$$

That is, in this case, $\mathbb{E}[\tau_1(\mathcal{C})] = k$.

The next theorem extends Example 1 to any code $\mathcal{C}$ and an information strand $\boldsymbol{x}_i$ with exactly two minimal retrieval sets A, B such that $A \cap B = \emptyset$.

Theorem 8. Let $\mathcal{C}$ be an (n, k) code and $i \in [k]$. If $\mathcal{D}(i) = \{A, B\}$, for two disjoint retrieval sets, $A \cap B = \emptyset$, then $\mathbb{E}[\tau_i(\mathcal{C})] = n \cdot \left(H_{|A|} + H_{|B|} - H_{|A|+|B|} \right).$

A direct corollary from Theorem 8 is that Example 1 can be generalized to any $(n = k + 1, k)$ simple parity code $\mathcal{C}$, and for any $i \in [k]$ the expected number of draws to retrieve $\boldsymbol{u}_i$ using $\mathcal{C}$ is exactly k.

Corollary 3. Assume $\mathcal{C}$ is the $(n = k + 1, k)$ simple parity code (i.e., $\mathbf{X} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k, \sum_{j=1}^{k} \boldsymbol{u}_j)$). Then, for any $i \in [k]$, we have that, $\mathbb{E}[\tau_i(\mathcal{C})] = k$ and $T_{\max}^{\mathcal{C}} = T_{\text{avg}}^{\mathcal{C}} = k$.

The proof of Theorem 8 relies on the inclusion-exclusion principle and can be found in Appendix I. The proof can also be extended to more than two retrieval sets. Since the extension is technical and repeats the same ideas as the ones from Theorem 8, it is omitted from the paper.

Corollary 4. Let $\mathcal{C}$ be an (n, k) code and $i \in [k]$. If $\mathcal{D}(i) = \{A_1, A_2, \ldots, A_v\}$ for mutually disjoint retrieval sets, then

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

Corollary 3 states that the simple parity code does not improve the value of $T_{\max}^{\mathcal{C}}$. 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 2, if no code is used, then we have that $T_{\max} = T_{\text{avg}} = \mathbb{E}[\tau_i] = k$ for any $i \in [k]$. On the other hand, assume $\mathcal{C}$ is a k-*non systematic MDS code* in which the minimal size of a retrieval set, for each of the information strands is k. In other words, any set of less than k encoded strands is not a retrieval set. If $\mathcal{C}$ is used, then in order to retrieve any specific information strand, one should sample a subset of k distinct encoded strands. Hence, by Corollary 1, for any $i \in [k]$, we have that, $T_{\max}^{\mathcal{C}} = \mathbb{E}[\tau_i(\mathcal{C})] = \sum_{j=0}^{k-1} \frac{n}{n-j} \approx n \log(\frac{n}{n-k})$, while if $\frac{k}{n} = R$ is a constant, we have that $n \log(\frac{n}{n-k}) = \frac{k}{R} \log(\frac{1}{1-R}) > k$. The next theorem discusses the case where $\mathcal{C}$ is a systematic MDS code and shows that for any such code the expected sample size is exactly k. The proof can be found in Appendix J.

Theorem 9. Let $\mathcal{C}$ be a systematic $[n, k]$ MDS code. For any $i \in [k]$ we have that $\mathbb{E}[\tau_i(\mathcal{C})] = k$ and hence $T_{\max}^{\mathcal{C}} = T_{\text{avg}}^{\mathcal{C}} = k$.

C. Reducing the Singleton Coverage Depth Below 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 codes do not improve upon the case where no coding is used. Next, we present families of (n, k) codes for which $T_{\max}^{\mathcal{C}} < k$. We start with the following example of an $(8, 4)$ code.

Example 2. Let $\mathcal{C}_{(8,4)}$ be the $(8,4)$ code defined as follows. Let $\mathbf{U}_{(8,4)} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3, \boldsymbol{u}_4) \in (\Sigma^\ell)^4$ and let

$$\mathbf{X}_{(8,4)} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3, \boldsymbol{u}_4, \boldsymbol{u}_1 + \boldsymbol{u}_2, \boldsymbol{u}_2 + \boldsymbol{u}_3, \boldsymbol{u}_3 + \boldsymbol{u}_4, \boldsymbol{u}_4 + \boldsymbol{u}_1) \in (\Sigma^\ell)^8.$$

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

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

Let $\mathcal{E}_{r-1}$ be the random variable that represents 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

$$P\left[\tau_1(\mathcal{C}_{(8,4)}) \geq r\right]$$
$$= \sum_{i=1}^{5} P\left[\tau_1(\mathcal{C}_{(8,4)}) \geq r | \mathcal{E}_{r-1} = i\right] \cdot P\left[\mathcal{E}_{r-1} = i\right]$$
$$+ P\left[\tau_1(\mathcal{C}_{(8,4)}) \geq r | \mathcal{E}_{r-1} \geq 6\right] \cdot P\left[\mathcal{E}_{r-1} \geq 6\right]$$
$$= \sum_{i=1}^{5} P\left[\tau_1(\mathcal{C}_{(8,4)}) \geq r | \mathcal{E}_{r-1} = i\right] \cdot P\left[\mathcal{E}_{r-1} = i\right].$$

It can be readily verified that $P\left[\tau_1(\mathcal{C}_{(8,4)}) \geq r | \mathcal{E}_{r-1} = 1\right] = \frac{7}{8}$. In case $\mathcal{E}_{r-1} = 2$, there are $\binom{8}{2} = 28$ different pairs of strands, and since $\tau_1(\mathcal{C}_{(8,4)}) \geq r$, we should consider only the pairs from which $\boldsymbol{u}_1$ cannot be retrieved. Note that two of the pairs are in $\mathcal{D}(1)$ and 7 additional pairs contain $\boldsymbol{u}_1$. Hence we have that $P\left[\tau_1(\mathcal{C}_{(8,4)}) \geq r | \mathcal{E}_{r-1} = 2\right] = \frac{28-9}{28} = \frac{19}{28}$. Similarly, there are $\binom{8}{3} = 56$ different triples, from which $\binom{7}{2} = 21$ contain $\boldsymbol{u}_1$, five more triples contain $\{\boldsymbol{u}_2, \boldsymbol{x}_{1,2}\}$ and do not contain $\boldsymbol{u}_1$, additional five triples contain $\{\boldsymbol{u}_3, \boldsymbol{x}_{1,3}\}$ (and do not contain $\boldsymbol{u}_1$), and two more triples are in $\mathcal{D}(1)$. That is, $P\left[\tau_1(\mathcal{C}_{(8,4)}) \geq r | \mathcal{E}_{r-1} = 3\right] = \frac{56-21-5-5-2}{56} = \frac{23}{56}$. Using similar counting techniques, it can be shown that $P\left[\tau_1(\mathcal{C}_{(8,4)}) \geq r | \mathcal{E}_{r-1} = 4\right] = \frac{8}{70}$, and $P\left[\tau_1(\mathcal{C}_{(8,4)}) \geq r | \mathcal{E}_{r-1} = 5\right] = \frac{1}{56}$. 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 we obtain that

$$\mathbb{E}[\tau_1(\mathcal{C}_{(8,4)})] = \sum_{r=1}^{\infty} P\left[\tau_1(\mathcal{C}_{(8,4)}) \geq r\right]$$
$$= \frac{403}{105} \approx 3.838 = 0.9595k.$$

Example 2 can be extended to any integer $k \geq 2$ as follows.

Construction 1. Let $\mathcal{C}_{(2k,k)}$ be the $(n = 2k, k)$ code such that

$$\mathbf{U}_{(2k,k)} = (\boldsymbol{u}_1, \boldsymbol{u}_2, ..., \boldsymbol{u}_k) \in (\Sigma^\ell)^k$$

and

$$\mathbf{X}_{(2k,k)} = (\boldsymbol{u}_1, ..., \boldsymbol{u}_k, \boldsymbol{u}_1 + \boldsymbol{u}_2, ..., \boldsymbol{u}_{k-1} + \boldsymbol{u}_k, \boldsymbol{u}_k + \boldsymbol{u}_1) \in (\Sigma^\ell)^{2k}.$$

Similarly to Example 2, the value $\mathbb{E}[\tau_1(\mathcal{C}_{(2k,k)})]$ can be expressed using the conditional probabilities $P[\tau_1(\mathcal{C}_{(2k,k)}) \geq$

$r|\mathcal{E}_{r-1} = i]$. The evaluation of these conditional probabilities can be done using a recursive formula which is given in the next theorem together with the expected value of $\tau_1(\mathcal{C}_{(2k,k)})$, while the proof appears in Appendix J.

Theorem 10. For any $k \geq 2$, and any $j \in [k]$ we have that

$$\mathbb{E}[\tau_j(\mathcal{C}_{(2k,k)})] = 1 + \sum_{i=1}^{2k-3} B(k,i) \cdot \frac{2k}{(2k-i)\binom{2k}{i}},$$

where

$$B(k,i) = \begin{cases} \binom{2k-1}{i} + 2B(k-1,i-1) - B(k-2,i-2) & k \geq 2, i \geq 2 \\ 1 & k \geq 0, i = 0 \\ 2k+1 & k \geq 0, i = 1 \\ 1 & k = 1, i = 2 \\ 0 & k = 0, i \geq 2 \\ 0 & k = 1, i \geq 3 \end{cases}.$$

Even though we did not solve the recursive formula in Theorem 10 to obtain an exact value for $\mathbb{E}[\tau_1(\mathcal{C}_{(2k,k)})]$, we used it to calculate $\mathbb{E}[\tau_1(\mathcal{C}_{(2k,k)})]$ for values $2 \leq k \leq 100$ and the results can be found in Fig 5. Based on these results we have the following conjecture.

Conjecture 1. For any $k \geq 4$ and any $j \in [k]$, we have that $\mathbb{E}[\tau_j(\mathcal{C}_{(2k,k)})] < k$. Moreover, the ratio $\frac{\mathbb{E}[\tau_j(\mathcal{C}_{(2k,k)})]}{k}$ decreases with k and

$$\lim_{k \to \infty} \frac{\mathbb{E}[\tau_j(\mathcal{C}_{(2k,k)})]}{k} < 0.9456.$$

The following definition is used in the next theorem.

Definition 2. Given an (n,k) code $\mathcal{C}$ as defined above and an integer $\gamma \geq 1$, we say that a $(\gamma n, \gamma k)$ code $\mathcal{C}^\gamma$ is the γ-block code of $\mathcal{C}$ if for an information word

$$\mathbf{U} = \mathbf{U}_1 \circ \mathbf{U}_2 \cdots \circ \mathbf{U}_\gamma$$
$$= (\boldsymbol{u}_1, ..., \boldsymbol{u}_k) \circ (\boldsymbol{u}_{k+1}, ..., \boldsymbol{u}_{2k}) \circ \cdots \circ (\boldsymbol{u}_{(\gamma-1)k+1}, ..., \boldsymbol{u}_{\gamma k}),$$

the corresponding codeword $\mathcal{X}_\gamma$ satisfies,

$$E_{\mathcal{C}^\gamma}(\mathbf{U}) = \mathbf{X} = \mathbf{X}_1 \circ \mathbf{X}_2 \circ \cdots \circ \mathbf{X}_\gamma$$
$$= E_{\mathcal{C}}(\mathbf{U}_1) \circ E_{\mathcal{C}}(\mathbf{U}_2) \circ \cdots \circ E_{\mathcal{C}}(\mathbf{U}_\gamma),$$

where $E_{\mathcal{C}}$ denotes the encoder of the code $\mathcal{C}$.

In the next theorem, we show that given an (n,k) code $\mathcal{C}$, one can increase k by using a γ-block code $\mathcal{C}^\gamma$, without changing the ratio between the expected number of draws to the number of information strands; see Appendix L for the proof.

Theorem 11. Let $\mathcal{C}$ be an (n,k) code. For an integer $\gamma \geq 1$, let $\mathcal{C}^\gamma$ be a γ-block code of $\mathcal{C}$. For any $1 \leq i \leq \gamma k$, it holds that, $\mathbb{E}[\tau_i(\mathcal{C}^\gamma)] = \gamma \mathbb{E}[\tau_{i'}(\mathcal{C})]$, where $i' \equiv i \pmod{k}$ and $1 \leq i' \leq k$.

Theorem 11 implies that given an (n,k) code $\mathcal{C}$, that achieves good results in terms of minimizing the expressions $\frac{\mathbb{E}[\tau_i(\mathcal{C})]}{k}$, for $i \in [k]$, it is possible to construct an infinite family of fixed-rate codes $\{\mathcal{C}^\gamma\}_{\gamma=1}^{\infty}$, such that for any integer $\gamma \geq 1$,

$\mathcal{C}^\gamma$ is an $(\gamma n, \gamma k)$ code and for any $i_\gamma \in [\gamma k]$ there exists $i \in [k]$, such that

$$\frac{\mathbb{E}[\tau_{i_\gamma}(\mathcal{C}^\gamma)]}{\gamma k} = \frac{\mathbb{E}[\tau_i(\mathcal{C})]}{k}.$$

That is, for any integer $\gamma \geq 1$, the code $\mathcal{C}^\gamma$ has the *same behavior* as the code $\mathcal{C}$ in terms of minimizing the normalized expected singleton coverage depth. Hence, combining Example 2 and Theorem 11 leads to the following corollary.

Corollary 5. For any integer $\gamma \geq 1$, let $\mathcal{C}^\gamma_{(8\gamma, 4\gamma)}$ be the γ-block code of $\mathcal{C} = \mathcal{C}_{(8,4)}$ (see Example 2). For any $i_\gamma \in [\gamma k]$, where $k = 4$, we have that

$$\mathbb{E}\left[\tau_{i_\gamma}(\mathcal{C}^\gamma_{(8\gamma, 4\gamma)})\right] = T^{\mathcal{C}^\gamma_{(8\gamma, 4\gamma)}}_{\max} = T^{\mathcal{C}^\gamma_{(8\gamma, 4\gamma)}}_{\mathrm{avg}} = 0.9595\gamma k.$$

Note that our numerical computations of the expression in Theorem 10 imply that the value $\frac{\mathbb{E}[\tau_i(\mathcal{C}_{(2k,k)})]}{k}$ decreases with k, for $2 \leq k \leq 100$. In particular, for $k > 3$, $\frac{\mathbb{E}[\tau_i(\mathcal{C}_{(2k,k)})]}{k} \leq 0.9456$ and thus by Theorem 11 it is possible to construct codes that improve upon the result in Corollary 5 for infinite values of k.

Next, we demonstrate that the value $\frac{\mathbb{E}[\tau_i(\mathcal{C})]}{k}$ can be further reduced, by letting the rates of our codes vanish.

Construction 2. Let n be an integer, $p \in (0,1)$, and assume for simplicity that np is an integer that is dividable by k. Additionally, let $\mathcal{C}^{\mathrm{MDS}}_{n,k,p}$ be a $[n(1-p)+k, k]$ systematic MDS code. We define the (n,k) code $\mathcal{C}^k_{n,p}$ as follows. For $\mathbf{U} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k) \in (\Sigma^\ell)^k$, let

$$(\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{(1-p)n}) \in (\Sigma^\ell)^{n(1-p)+k}$$

be the encoding of $\mathbf{U}$ using the encoder of $\mathcal{C}^{\mathrm{MDS}}_{n,k,p}$. Then,

$$\mathbf{X} = (\underbrace{\boldsymbol{u}_1, ..., \boldsymbol{u}_1}_{\frac{pn}{k}\text{ times}}, \underbrace{\boldsymbol{u}_2, ..., \boldsymbol{u}_2}_{\frac{pn}{k}\text{ times}}, ..., \underbrace{\boldsymbol{u}_k, ..., \boldsymbol{u}_k}_{\frac{pn}{k}\text{ times}}, \boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_{(1-p)n}),$$

and $\mathbf{X} \in (\Sigma^\ell)^n$.

Theorem 12. For $k = 2, 3$, there exists $p_2, p_3 \in (0,1)$, such that for any information strand $i \in [k]$, we have that, $\mathbb{E}[\tau_i(\mathcal{C}^2_{n,p_2})] \approx 1.83 = 0.9143k$, and $\mathbb{E}[\tau_i(\mathcal{C}^3_{n,p_3})] \approx 2.67 = 0.89k$.

The proof of Theorem 12 can be found in Appendix M. Combining Theorem 11 and Theorem 12 leads to the following corollary.

Corollary 6. Let $p_2, p_3 \in (0,1)$ be the constants from Theorem 12. For any integer $\gamma \geq 1$, let $\mathcal{C}^{2,\gamma}_{n,p_2}$ be the $(n\gamma, 2\gamma)$ γ-block code of $\mathcal{C} = \mathcal{C}^2_{n,p_2}$, and similarly let $\mathcal{C}^{3,\gamma}_{n,p_3}$ be the $(n\gamma, 3\gamma)$ γ-block code of $\mathcal{C} = \mathcal{C}^3_{n,p_3}$ (see Definition 2). For any $i_2 \in [2\gamma]$, and $i_3 \in [3\gamma]$ we have that

$$\mathbb{E}[\tau_{i_2}(\mathcal{C}^{2,\gamma}_{n,p_2})] = T^{\mathcal{C}^{2,\gamma}_{n,p_2}}_{\max} = T^{\mathcal{C}^{2,\gamma}_{n,p_2}}_{\mathrm{avg}} \approx 1.83\gamma = 0.9143 \cdot (2\gamma),$$

and

$$\mathbb{E}[\tau_{i_3}(\mathcal{C}^{3,\gamma}_{n,p_3})] = T^{\mathcal{C}^{3,\gamma}_{n,p_3}}_{\max} = T^{\mathcal{C}^{3,\gamma}_{n,p_3}}_{\mathrm{avg}} \approx 2.67\gamma = 0.89 \cdot (3\gamma).$$

The evaluation of $\mathbb{E}[\tau_i(\mathcal{C}^k_{n,p})]$ for $k > 3$ can be done using the same technique, however, it becomes less elegant and we do not attempt to evaluate the latter expression rigorously.

Nevertheless, we did try to gain a better understanding of the behavior of these codes by computer simulations as follows. Each simulation was done while fixing $k \in \{1, \ldots, 10\} \cup \{20, 30, \ldots 100\}$ and $p \in \{0.2, 0.4, 0.6, 0.8\}$. For each pair of k and p we started by encoding k information strands with an $[\lfloor(1-p)n\rfloor + k, k]$ MDS code $\mathcal{C}^{\mathrm{MDS}}_{n,k,p}$, from which we constructed the (n,k) code $\mathcal{C}^k_{n,p}$ (see Construction 2). Then, we simulated the sampling process by picking a single strand at each draw (with an equal probability of $\frac{1}{n}$). The simulation stops whenever we can recover $\boldsymbol{u}_1$. We repeated this process 10^7 times for each pair of k and p and plotted the mean number of the required draws, which is an empirical approximation of $\mathbb{E}[\tau_i(\mathcal{C}^k_{n,p})]$. Our simulations imply that for most of the tested values of k, the optimal value of p is around 0.6. Furthermore, it can be seen that $\mathbb{E}[\tau_i(\mathcal{C}^k_{n,p})]$ decreases as k increases. Finally, it should be noted that even though such codes are not applicable, they allow us to gain insights about the achievable values of $\mathbb{E}[\tau_1(\mathcal{C})]$.

D. Lower Bounds

This section concludes with lower bounds on the value of $\mathbb{E}[\tau_i(\mathcal{C})]$.

Lemma 3. For any (n,k) code $\mathcal{C}$, $T^{\mathcal{C}}_{\max} \geq \frac{k+1}{2}$.

Proof: Assume the word $\mathbf{U}$ was encoded to the codeword $\mathbf{X}$. Every sequence of reads can be expressed as a vector $\boldsymbol{v} \in [n]^*$, and for every such a $\boldsymbol{v}$, denote by $n_i(\boldsymbol{v})$, for $i \in [k]$, the minimum read index h which allows retrieving the i-th information strand $\boldsymbol{u}_i$. The key intuition behind our approach is that each new sample collected during the sequence of reading the strands allows us to recover at most one new information strand. Hence,

$$\sum_{i=1}^k n_i(\boldsymbol{v}) = n_1(\boldsymbol{v}) + n_2(\boldsymbol{v}) + \cdots + n_k(\boldsymbol{v}) \geq \sum_{i=1}^k i = k(k+1)/2.$$

Hence, it follows that, $\sum_{i=1}^k \tau_i(\mathcal{C}) \geq k(k+1)/2$ and therefore

$$\mathbb{E}\left[\sum_{i=1}^k \tau_i(\mathcal{C})\right] = \sum_{i=1}^k \mathbb{E}[\tau_i(\mathcal{C})] \geq k(k+1)/2.$$

In particular, there exists $i \in [k]$ for which $\mathbb{E}[\tau_i(\mathcal{C})] \geq \frac{k+1}{2}$, i.e., $T^{\mathcal{C}}_{\max} \geq \frac{k+1}{2}$. ∎

Even though the bound in Lemma 3 holds for any code $\mathcal{C}$, it appears that in most cases the bound is not tight. To obtain a tighter lower bound on $T^{\mathcal{C}}_{\max}$ in the next theorem we also consider the rate of the code $\mathcal{C}$.

Theorem 13. Let $\mathcal{C}$ be an (n,k) code. It holds that

$$T^{\mathcal{C}}_{\max} \geq \frac{n}{k} \cdot \sum_{i=0}^k \frac{k-i}{n-i} = n - \frac{n(n-k)}{k} \cdot (H_n - H_{n-k}).$$

Proof: Let us use the same notations as in the proof of Lemma 3. Additionally, denote by $t_i(v)$ the time to collect the i-th new sample (after collecting the previous one). Clearly, we have that

$$\sum_{i=1}^k \tau_i(\mathcal{C}) = \sum_{i=1}^k n_i(v) \geq \sum_{i=1}^k \sum_{j=1}^i t_j(v).$$
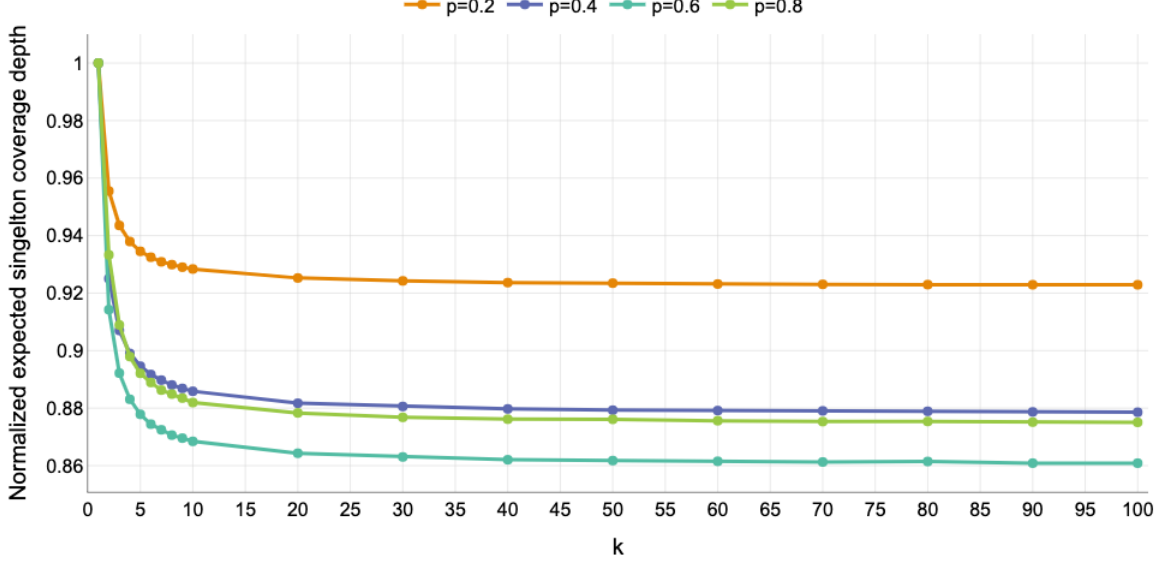
Fig. 3: Approximated values of $\mathbb{E}[\tau_i(\mathcal{C}_{n,p}^k)]$ (Construction 2) for different values of $p \in \{0.2, 0.4, 0.6, 0.8\}$ as a function of $k \in \{1, 2, \ldots, 10\} \cup \{20, 30, \ldots, 100\}$, where $n = 10^8$. The approximated values were obtained empirically by $10,000,000$ computer simulations per any pair of values of k and p. The presented results are normalized by k.

Define $t_j(\mathcal{C})$ to be the random variable that governs the time to collect the j-th new sample (after collecting the previous one). Hence,

$$\sum_{i=1}^{k} \mathbb{E}\left[\tau_i(\mathcal{C})\right] = \mathbb{E}\left[\sum_{i=1}^{k} \tau_i(\mathcal{C})\right]$$
$$\geq \mathbb{E}\left[\sum_{i=1}^{k} \sum_{j=1}^{i} t_j(\mathcal{C})\right]$$
$$= \sum_{i=1}^{k} \sum_{j=1}^{i} \mathbb{E}\left[t_j(\mathcal{C})\right].$$

Note that for any $j \in [k]$, we have that $t_j(\mathcal{C})$ is a geometric random variable with success probability $p_j = \frac{n-(j-1)}{n}$ and so $\mathbb{E}[t_j(\mathcal{C})] = \frac{n}{n-(j-1)}$, and

$$\sum_{i=1}^{k} \mathbb{E}\left[\tau_i(\mathcal{C})\right] \geq \sum_{i=1}^{k} \sum_{j=1}^{i} \mathbb{E}\left[t_j(\mathcal{C})\right]$$
$$= \sum_{i=1}^{k} \sum_{j=1}^{i} \frac{n}{n-(j-1)}$$
$$= n \sum_{i=1}^{k} \left(\frac{1}{n-i+1} + \frac{1}{n-i+2} + \ldots + \frac{1}{n}\right)$$
$$= n \left(\frac{k}{n} + \frac{k-1}{n-1} + \ldots + \frac{1}{n-k+1}\right)$$
$$= n \sum_{i=0}^{k-1} \frac{k-i}{n-i}.$$

For any $i \in [k]$, we have that

$$\frac{k-i}{n-i} = \frac{k}{n} - \left(1 - \frac{k}{n}\right) \frac{i}{n-i},$$

which implies that

$$n \sum_{i=0}^{k} \frac{k-i}{n-i} = n \sum_{i=0}^{k-1} \left(\frac{k}{n} - \left(1 - \frac{k}{n}\right) \frac{i}{n-i}\right)$$
$$= k^2 - n \left(1 - \frac{k}{n}\right) \sum_{i=0}^{k-1} \frac{i}{n-i}$$
$$= k^2 - (n-k) \sum_{i=0}^{k-1} \left(\frac{n}{n-i} - 1\right)$$
$$= k^2 + k(n-k) - n(n-k) \sum_{i=0}^{k-1} \frac{1}{n-i}$$
$$= nk - n(n-k)(H_n - H_{n-k}).$$

Hence we have that

$$\frac{1}{k} \sum_{i=1}^{k} \mathbb{E}\left[\tau_i(\mathcal{C})\right] \geq \frac{n}{k} \cdot \sum_{i=0}^{k} \frac{k-i}{n-i}$$
$$= n - \frac{n(n-k)}{k} \cdot (H_n - H_{n-k}).$$

In particular, there exists $i \in [k]$ for which $\mathbb{E}[\tau_i(\mathcal{C})] \geq \frac{n}{k} \cdot \sum_{i=0}^{k} \frac{k-i}{n-i} = n - \frac{n(n-k)}{k} \cdot (H_n - H_{n-k})$, i.e., $T_{\max}^{\mathcal{C}} \geq \frac{n}{k} \cdot \sum_{i=0}^{k} \frac{k-i}{n-i} = n - \frac{n(n-k)}{k} \cdot (H_n - H_{n-k})$. ∎

Lastly, we conclude with the following lemma, which is proven in Appendix N.

Corollary 7. Let $0 < R < 1$ and consider a sequence of codes $\{\mathcal{C}_i\}_{i=1}^{\infty}$ with parameters (n_i, k_i) such that for any i, $n_i < n_{i+1}$, and $R = \frac{k_i}{n_i}$. It holds that,

$$\lim_{i \to \infty} \frac{T_{\max}^{\mathcal{C}_i}}{k_i} \geq \left(\frac{1}{R} + \frac{1-R}{R^2} \cdot \log(1-R) \right).$$

That is, for any $\varepsilon > 0$, there exists i large enough (i.e., n_i, k_i large enough) such that,

$$T_{\max}^{\mathcal{C}_i} \geq k_i \left(\frac{1}{R} + \frac{1-R}{R^2} \cdot \log(1-R) \right) - \varepsilon.$$

Fig. 4 presents a comparison between the lower bounds of Lemma 3 and Corollary 7 as a function of the code rate $R = \frac{k}{n}$. As can be seen in the figure, in most cases, the bound in Corollary 7 is tighter than the one from Lemma 3. More than that, the code rate from which the bound in Corollary 7 is tighter than the bound from Lemma 3 decreases with k.

Finally, we give in Fig. 5 a comparison of the normalized expected singleton coverage depth for different codes with rate of exactly $R = 0.5$. It can be seen in the figure that the k-non systematic MDS code achieves the worst results, while the code in Theorem 10 achieves the best results, which are roughly 55% lower than the k-non systematic MDS code and roughly 10% lower than a systematic MDS code. To offer a better understanding of these results, the lower bounds discussed in Lemma 3 and Corollary 7 are also given in the figure.

VII. Conclusions and Discussion

In this paper, we have introduced and extensively investigated the novel problem of DNA coverage depth, aiming to reduce sequencing costs and latency while ensuring high-accuracy retrieval. Our contributions encompass the MDS coverage depth problem, demonstrating the superiority of MDS codes in the noiseless channel. For noisy channels, we proved several bounds on the probability of successfully retrieving the information for a given sample size. Additionally, we have explored the singleton coverage depth problem, revealing insights into code properties and retrieval times, as well as presenting code constructions that can improve the retrieval time. These findings collectively provide a foundational framework for designing efficient and reliable DNA storage systems, with potential implications for advancing the field.

Nonetheless, future research should address the diverse challenges posed by different noise models, investigate coding schemes beyond MDS codes, and extend the coverage depth problem for additional scenarios. Several possible directions and open problems are listed below.

1) Extend the results presented in this paper with respect to Problem 1 from a uniform distribution to additional channel distributions $\boldsymbol{p}$; e.g. the normal distribution.
2) In this work, the noisy channel was modeled by a parameter t, under the assumption that retrieval succeeds with probability 1 given t or more noisy copies and fails otherwise. A very relevant extension to this noise model is to consider the more realistic behavior of the channel, in which the success probability can increase or decrease

as a function of the number of noisy copies, i.e., as a function of the cluster size.
3) Define and study the coverage depth random access problem for the case in which a subset of size greater than one should be retrieved. This can be considered for arbitrary subsets of the information strands or for pre-defined subsets, that represent units of information (e.g. files).
4) Study the coverage depth random access problem under the assumption of noisy channel and/or channel with non-uniform distribution.
5) As mentioned in Section III-B, the problems studied in this paper are strongly related to the problem of optimizing the ratio of writing and reading costs, which was discussed in [7]. In particular, the costs of the reading and writing process in this setup are considered somewhat equivalent, while in this work, we neglect the writing cost. As archival storage is considered one of the most promising applications for DNA storage in the near future, it makes sense to put less focus on reducing the writing cost. However, this cost should still be considered, and the framework of the problem should be extended to incorporate this idea. Such extension can be done, for example, by a parameter that captures the number of times a piece of information is expected to be accessed. We note that the idea for this extension follows from a comment of one of the anonymous reviewers.

Recently, following the publication of the conference version of this work, several works have been studied as extensions of the model given in this paper. The results on the singleton random access problem were extended in [17]. Additionally, several results on the random access problem for the case where we are interested in a subset of the information strands (i.e., Open Problem 3) were presented in [1]. The non-random access scenario of the DNA coverage depth problem was further extended in [30], [36] to support the setup of the combinatorial composite of DNA shortmers [31].

Appendix A

Proof of Theorem 2:

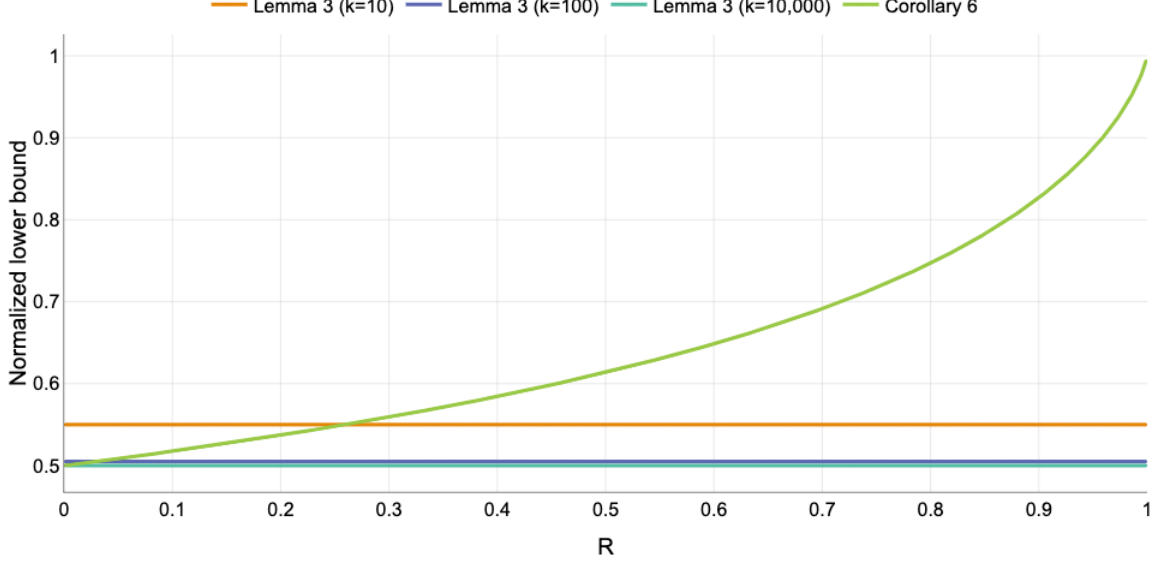If $0 < R < 1$ is fixed, then n goes to infinity together with k and thus we have that,

Fig. 4: Comparison of the lower bounds (Lemma 3 and Corollary 7) as a function of the rate $R = \frac{k}{n}$. The presented results are normalized by k.
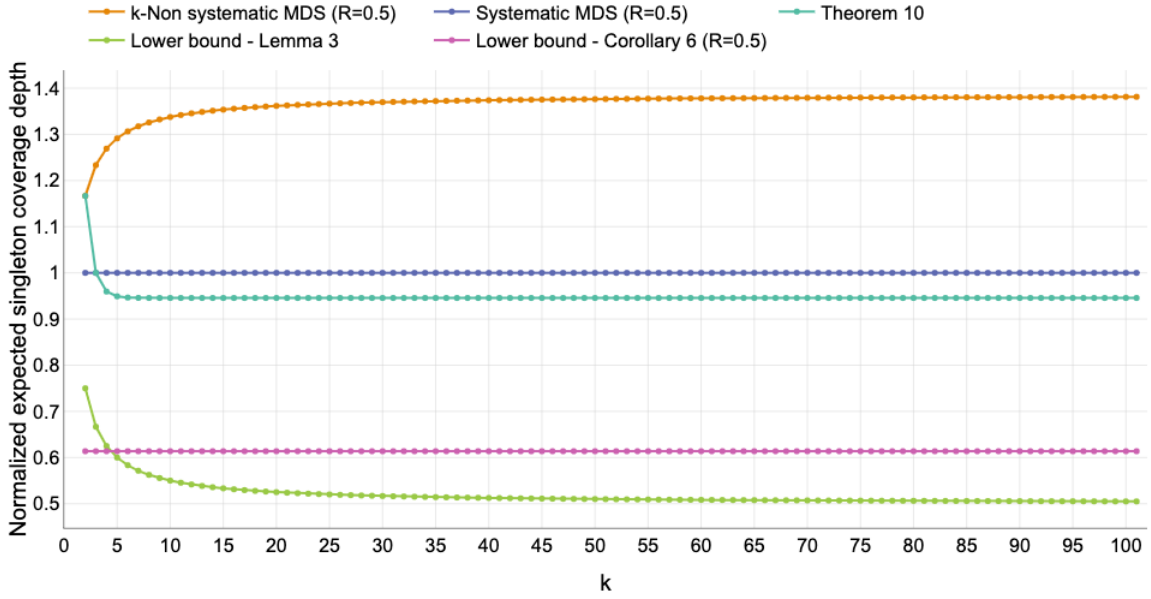

Fig. 5: Comparison of the normalized expected singleton coverage depth for code with rate $R = 0.5$.

below holds.

$$
\lim_{n \to \infty} \frac{\mathbb{E}[\nu_1(n, k = \lfloor nR \rfloor)]}{k} = \lim_{n \to \infty} \frac{n(H_n - H_{n-k})}{k}
$$
$$
= \frac{1}{R} \log \left(\frac{1}{1 - R} \right),
$$

where the first equality holds from [15] and the second equality is a known result.

Furthermore, in the case in which we have a sequence of MDS codes $\{\mathcal{C}_i\}_{i=1}^{\infty}$, such that $\lim_{i \to \infty} \frac{k_i}{n_i} = 0$, the equality

$$
\lim_{i \to \infty} \frac{\mathbb{E}[\nu_1(n_i, k_i)]}{k_i} = \lim_{i \to \infty} \frac{n_i(H_{n_i} - H_{n_i - k_i})}{k_i}
$$
$$
= \lim_{i \to \infty} \frac{\sum_{j=0}^{k_i - 1} \frac{n_i}{n_i - j}}{k_i},
$$

where,

$$\lim_{i\to\infty} \frac{\sum_{j=0}^{k_i-1} \frac{n_i}{n_i-j}}{k_i} \le \lim_{i\to\infty} \frac{k_i\left(\frac{n_i}{n_i-(k_i-1)}\right)}{k_i}$$

$$= \lim_{i\to\infty} \frac{k_i\left(\frac{1}{1-\frac{k_i-1}{n_i}}\right)}{k_i} = 1,$$

and,

$$\lim_{i\to\infty} \frac{\sum_{j=0}^{k-1} \frac{n_i}{n_i-j}}{k_i} \ge \lim_{i\to\infty} \frac{k_i\left(\frac{n_i}{n_i-0}\right)}{k_i} = 1.$$

Thus, we can conclude that, $\lim_{i\to\infty} \frac{\mathbb{E}[\nu_1(n_i,k_i)]}{k_i} = 1$. ∎

APPENDIX B

Claim 6. For $n > 16$, it holds that,

$$\sum_{j=0}^{t-1} \binom{r}{j}\left(\frac{1}{n}\right)^j\left(1-\frac{1}{n}\right)^{r-j} \le t\binom{r}{t-1}\left(\frac{1}{n}\right)^{t-1}\left(1-\frac{1}{n}\right)^{r-(t-1)}$$

Proof: To prove the claim, we show that for $0 \le j \le t-1$,

$$\binom{r}{j}\left(\frac{1}{n}\right)^j\left(1-\frac{1}{n}\right)^{r-j} \le \binom{r}{j+1}\left(\frac{1}{n}\right)^{j+1}\left(1-\frac{1}{n}\right)^{r-(j+1)}.$$

The latter can be proved by showing the following equivalent inequality

$$n \le \frac{r-j}{j+1}\left(1-\frac{1}{n}\right)^{-1}.$$

The expression $\frac{r-j}{j+1}\left(1-\frac{1}{n}\right)^{-1}$ is minimized at $j = t-1$ (considering only $j \in \{0,1,\dots,t-1\}$), thus it is enough to show that, $n \le \frac{r-t+1}{t}\left(1-\frac{1}{n}\right)^{-1} = \frac{r-t+1}{t}\left(\frac{n}{n-1}\right)$, which follows if

$$r = r(n,k,t) \ge t(n-1) + (t-1).$$

Lastly, it can be verified that $r(n,k,t) \ge t(n-1) + (t-1)$ for any integers $n > 16$ and $t > 1$. ∎

Proof of Theorem 4:

Denote $r_f \triangleq r_f(n,k,t)$. Similarly to the proof of Theorem 3, when n is large enough, the probability that urn i has at most $t-1$ balls after r_f draws is denoted by $P(z_i(n,r_f) \le t-1)$, where $z_i(n,r_f)$ is a defined as in the proof of Theorem 3. Thus, the probability is given by,

$$P(z_i(n,r_f) \le t-1) \le 3t\cdot\left(\frac{r_f}{n}\right)^{t-1}\left(1-\frac{1}{n}\right)^{n\left(\frac{r_f}{n}-\frac{t-1}{n}\right)}$$

$$\le 6t^t\frac{(2\cdot f(n))^{t-1}}{e^{t\cdot f(n)}}\cdot(1-R)$$

Now let us define a random variable Y as the number of urns with less than t balls. As in the proof of Theorem 3, we have that,

$$P(E_t^{(r_f)}) = P(Y \ge n-k+1) \le 6t^t\frac{(2\cdot f(n))^{t-1}}{e^{t\cdot f(n)}}(1-R),$$

where the last inequality follows from Markov's inequality. ∎

APPENDIX C

Proof of Lemma 1:

We first highlight that $\nu_t(n,k) \ge \nu_1(n,k)$, and thus it is enough to show that

$$P\left[\nu_1(n,k) \le n\log(\frac{n}{n-k}) - nc\right] \le e^{-c}\left(1 + \frac{1}{n-k}\right).$$

We have that,

$$e^{\log(\frac{n}{n-k})-c}\cdot\mathbb{E}\left[e^{-\frac{\nu_1(n,k)}{n}}\right]$$

$$= e^{\log(\frac{n}{n-k})-c}\cdot\sum_{j=1}^{\infty} e^{-\frac{j}{n}}P\left[\nu_1(n,k) = j\right]$$

$$= \sum_{j=1}^{\infty} e^{\log(\frac{n}{n-k})-c-\frac{j}{n}}P\left[\nu_1(n,k) = j\right]$$

$$= \sum_{j=1}^{\lfloor n\log(\frac{n}{n-k})-nc\rfloor} e^{\log(\frac{n}{n-k})-c-\frac{j}{n}}P\left[\nu_1(n,k) = j\right]$$

$$+ \sum_{j=1+\lfloor n\log(\frac{n}{n-k})-nc\rfloor}^{\infty} e^{\log(\frac{n}{n-k})-c-\frac{j}{n}}P\left[\nu_1(n,k) = j\right]$$

$$\ge \sum_{j=1}^{\lfloor n\log(\frac{n}{n-k})-nc\rfloor} e^{\log(\frac{n}{n-k})-c-\frac{j}{n}}P\left[\nu_1(n,k) = j\right]$$

$$\ge \sum_{j=1}^{\lfloor n\log(\frac{n}{n-k})-nc\rfloor} 1\cdot P\left[\nu_1(n,k) = j\right]$$

$$\ge P\left[\nu_1(n,k) \le n\log\left[\frac{n}{n-k}\right] - nc\right].$$

From [29], the generating function of the geometric random variable $\nu_1(n,k)$ is given by

$$G_{\nu_1(n,k)}(x) = \mathbb{E}[x^{\nu_1(n,k)}]$$

$$= \sum_{j=0}^{\infty} P[\nu_1(n,k) = j]x^j = \prod_{i=1}^{k}\frac{(n-(i-1))x}{n-(i-1)x}$$

$$= \prod_{i=1}^{k}\frac{(1-\frac{i-1}{n})x}{1-\frac{i-1}{n}x}.$$

∎

Thus, given $x = e^{-1/n}$, we get that,

$$\mathbb{E}\left[e^{\frac{-\nu_1(n,k)}{n}}\right] = \prod_{i=1}^{k}\frac{(1-\frac{i-1}{n})e^{-\frac{1}{n}}}{1-(\frac{i-1}{n})e^{-\frac{1}{n}}}$$

$$= \prod_{i=1}^{k}\frac{1-\frac{i-1}{n}}{e^{\frac{1}{n}}-\frac{i-1}{n}} \le \prod_{i=1}^{k}\frac{1-\frac{i-1}{n}}{1+\frac{1}{n}-\frac{i-1}{n}}$$

$$= \prod_{i=1}^{k}\frac{1-\frac{i-1}{n}}{1-\frac{i-2}{n}} = \frac{1-\frac{k-1}{n}}{1+\frac{1}{n}}$$

$$= \frac{n-k+1}{n+1} \le \frac{n-k+1}{n},$$

where in the first inequality we used the fact the $e^{\frac{1}{n}} \geq 1 + \frac{1}{n}$. Hence, it holds that, for positive c,

$$e^{\log(\frac{n}{n-k})-c} \cdot \mathbb{E}\left[e^{-\nu_1(n,k)/n}\right] \leq e^{-c} \cdot \frac{n}{n-k} \cdot \frac{n-k+1}{n}$$
$$= e^{-c}\left(1 + \frac{1}{n-k}\right).$$

Finally, we conclude that,

$$P\left[\nu_1(n,k) \leq n\log(\frac{n}{n-k}) - nc\right]$$
$$\leq e^{\log(\frac{n}{n-k})-c} \cdot \mathbb{E}\left[e^{-\nu_1(n,k)/n}\right]$$
$$\leq e^{-c}\left(1 + \frac{1}{n-k}\right).$$

APPENDIX D

Proof of Theorem 5:

First, we highlight that for any integer $t > 0$, it holds that $\nu_t(n,k) \geq \nu_1(n,k)$. Next, we recall the known results proven in [16], where they showed $\mathbb{E}[\nu_1(n,k)] = n(H_n - H_{n-k})$. Hence, we can conclude that the following holds for n large enough,

$$\mathbb{E}[\nu_t(n,k)] \geq \mathbb{E}[\nu_1(n,k)] = n(H_n - H_{n-k})$$
$$= n\left(\log(n) + \gamma + \frac{1}{2n} - \sum_{h=1}^{\infty} \frac{B_{2h}}{2hn^{2h}} - \log(n-k)\right.$$
$$\left. -\gamma - \frac{1}{2(n-k)} + \sum_{h=1}^{\infty} \frac{B_{2h}}{2h(n-k)^{2h}}\right)$$
$$= n\log\left(\frac{n}{n-k}\right) + \frac{1}{2}\left(1 - \frac{1}{1-R}\right)$$
$$- n\sum_{h=1}^{\infty} \frac{B_{2h}}{2hn^{2h}}\left(1 - \frac{1}{(1-R)^{2h}}\right)$$
$$= n\log\left(\frac{1}{1-R}\right) + nf_c(n,R),$$

where $\gamma \sim 0.5772156649$ is the Euler-Mascheroni constant, where the last equality was proven in [16]. Next, let $r_n \triangleq r(n,k,t)$ (recall that by (6), $r(n,k,t) = n\log(\frac{1}{1-R}) + nt\log\log(n) + 2n\log(t+1)$). In Theorem 3 we showed that $P[\nu_t(n,k) > r_n] < \varepsilon$. Using the same methods, in Appendix E we proved Theorem 14 which states that for any integer $i \geq 1$ and for n large enough, $P[\nu_t(n,k) > r_n \cdot i] < \varepsilon \cdot \frac{i^{t-1}}{\log^{t(i-1)}(n)}$,

and thus we can conclude that,

$$E[\nu_t(n,k)] = \sum_{r \in \mathbb{N}} P(\nu_t(n,k) \geq r)$$
$$= \sum_{r < r_n} P(\nu_t(n,k) \geq r) + \sum_{r \geq r_n} P(\nu_t(n,k) \geq r)$$
$$\leq 1 \cdot r_n + \sum_{r \geq r_n} P(\nu_t(n,k) \geq r)$$
$$= r_n + \sum_{i=1}^{\infty} \sum_{r=i \cdot r_n}^{(i+1) \cdot r_n} P(\nu_t(n,k) \geq r)$$
$$\leq r_n + \sum_{i=1}^{\infty} \sum_{r=i \cdot r_n}^{(i+1) \cdot r_n} P(\nu_t(n,k) \geq i \cdot r_n)$$
$$= r_n + \sum_{i=1}^{\infty} r_n \cdot P(\nu_t(n,k) \geq i \cdot r_n)$$
$$< r_n + \sum_{i=1}^{\infty} r_n \cdot \varepsilon \cdot \frac{i^{t-1}}{\log^{t(i-1)}(n)}$$
$$= r_n + \varepsilon \cdot r_n \sum_{i=1}^{\infty} \frac{i^{t-1}}{\log^{t(i-1)}(n)}$$
$$\overset{(a)}{<} r_n + 2\varepsilon \cdot r_n$$
$$= r_n \cdot (1 + 2\varepsilon),$$

where (a) follows since $\sum_{i=1}^{\infty} \frac{i^{t-1}}{\log^{t(i-1)}(n)} < 2$ for n large enough and any integer $t > 0$. Lastly, we simplify the expression,

$$\frac{1}{n}r_n(1 + 2\varepsilon)$$
$$= \left(\log\left(\frac{1}{1-R}\right) + t\log\log n + 2\log(t+1)\right)(1 + 2\varepsilon)$$
$$= \log\left(\frac{1}{1-R}\right) + \mathcal{O}(t\log\log n),$$

which completes the proof. ∎

APPENDIX E

Theorem 14. For any $\varepsilon > 0$, $n > e^{\frac{6t \cdot 2^{t-1}}{\varepsilon}} \geq 15$, and any integer $h \geq 1$, we have that,

$$P[\nu_t(n,k) > h \cdot r(n,k,t)] < \varepsilon \cdot \frac{h^{t-1}}{\log^{t(h-1)}(n)}.$$

Proof: Denote $r \triangleq r(n,k,t)$ and recall that within the context of the urn problem (see Section III-B), the random variable $\nu_t(n,k)$ 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 balls.

If r balls are drawn, we show that the probability that there are at least k urns each with at least t balls is approaching one when n grows. Analogous to the approach used in the previous section, we will show that if the number of balls thrown is at least r, then the probability to have at most $n-k+1$ urns which are *not* filled with t balls is approaching zero.

The approach leveraged in this section is inspired by a technique first employed by Erdős and Rényi in [12]. First, we define the following event.

$E_t^{(r)}$: After r rounds, there exists a set S_t, of $n - k + 1$ urns, each containing less than t balls.

Next, we show that the probability of $E_t^{(r)}$ approaches zero when n is large. We first define $z_i(n, r)$ for $1 \leq i \leq n$, as a random variable that governs the number of balls in the i-th urn, after r draws. For n large enough, the probability that urn i has at most $t - 1$ balls after $h \cdot r$ draws is denoted by $P(z_i(n, r) \leq t - 1)$ and is given by,

$$P(z_i(n, h \cdot r) \leq t - 1)$$
$$= \sum_{j=0}^{t-1} \binom{h \cdot r}{j} \left(\frac{1}{n}\right)^j \left(1 - \frac{1}{n}\right)^{h \cdot r - j}$$
$$\leq t \cdot \binom{h \cdot r}{t-1} \left(\frac{1}{n}\right)^{t-1} \left(1 - \frac{1}{n}\right)^{h \cdot r - (t-1)}$$
$$\leq t \cdot \left(\frac{h \cdot r \cdot e}{t-1}\right)^{t-1} \left(\frac{1}{n}\right)^{t-1} \left(1 - \frac{1}{n}\right)^{h \cdot r - (t-1)},$$

where the last inequality follows from the fact that $\binom{h \cdot r}{t-1} \leq \frac{hre}{t-1}^{t-1}$. Note that $\left(\frac{e}{t-1}\right)^{t-1} < 3$, for $t > 1$. Thus,

$$P(z_i(n, r) \leq t - 1) \leq 3t \left(\frac{hr}{n}\right)^{t-1} \left(1 - \frac{1}{n}\right)^{n\left(\frac{r}{n} - \frac{t-1}{n}\right)}.$$

We have that,

$$P(z_i(n, r) \leq t - 1)$$
$$\leq 3t \left(h \log\left(\frac{n}{n-k}\right) + ht \log\log(n) + 2h \log(t+1)\right)^{t-1} e^{\frac{-hr+t-1}{n}}$$
$$\leq 3t \cdot (2h \log n)^{t-1} \left(\frac{n-k}{n}\right)^h \left(\frac{1}{\log^{ht} n}\right) \left(\frac{1}{(t+1)^{2h}}\right) e^{\frac{t-1}{n}}$$
$$= 3t \cdot \frac{e^{\frac{t-1}{n}}}{(t+1)^{2h}} \cdot \frac{(2h \log n)^{t-1}}{\log^{ht}(n)} \cdot \left(\frac{n-k}{n}\right)^h$$
$$= 3t \cdot \frac{e^{\frac{t-1}{n}}}{(t+1)^{2h}} \cdot \frac{(2h)^{t-1}}{\log^{t(h-1)+1}(n)} \cdot \left(\frac{n-k}{n}\right)^h,$$

where the second inequality holds since for n large enough $h \log(\frac{n}{n-k}) + th \log\log(n) + 2h \log(t+1) \leq (2h \log n)$. It should be noted that for $n > t$, we have that $3t \cdot \frac{e^{\frac{t-1}{n}}}{(t+1)^{2h}} \leq 6t$, and hence,

$$P(z_i(n, r) \leq t - 1) \leq 6t \cdot \frac{(2h)^{t-1}}{\log^{t(h-1)+1}(n)} \cdot \left(\frac{n-k}{n}\right)^h.$$

Now let us define a random variable Y as the number of urns with less than t balls. From the linearity of expectation, regardless if the urns are independent or not, the expected number of urns that have less than t balls is,

$$\lim_{n \to \infty} \mathbb{E}[Y] = \lim_{n \to \infty} \Sigma_{i=1}^n \mathbb{E}[z_i(n, r)]$$
$$= \lim_{n \to \infty} nP(z_i(n, r) \leq t - 1)$$
$$\leq \lim_{n \to \infty} (n-k) \left(\frac{n-k}{n}\right)^{h-1} 6t \frac{(2h)^{t-1}}{\log^{t(h-1)+1}(n)}$$

Note that

$$P(E_{t+1}^{(r)}) = P(Y \geq n - k + 1)$$

Using Markov's inequality with $n - k + 1$ as the parameter we can conclude that,

$$P(Y \geq n - k + 1) \leq \left(\frac{n-k}{n}\right)^{h-1} 6t \cdot \frac{(2h)^{t-1}}{\log^{t(h-1)+1}(n)}.$$

Let us denote $\varepsilon^* = 6t \cdot \frac{2^{t-1}}{\log(n)}$, then we get that,

$$P(Y \geq n - k + 1) \leq \varepsilon^* \left(\frac{n-k}{n}\right)^{h-1} \cdot \frac{(h)^{t-1}}{\log^{t(h-1)}(n)}$$
$$\leq \varepsilon^* \cdot \frac{(h)^{t-1}}{\log^{t(h-1)}(n)}.$$

Hence, we get that $P(E_t^{(r)}) \to 0$ for n large enough which implies the statement in the theorem. ∎

APPENDIX F

In this appendix, we prove Theorem 6. The proof is partially based on Claim 2 which is proven next.

Proof of Claim 2:

Recall that by (8), for $r \geq nt$ we have that

$$E[X^{(r)}] \leq ne^{-(t-1)\log_2 \frac{n(t-1)}{r} - (r-(t-1))\log_2 \frac{(r-(t-1))n}{r(n-1)}}.$$

Hence, a sufficient condition for $E[X^{(r)}] \leq n - k = n(1 - R)$, it that

$$e^{-(t-1)\log_2 \frac{n(t-1)}{r} - (r-(t-1))\log_2 \frac{(r-(t-1))n}{r(n-1)}} \leq (1 - R). \quad (12)$$

Note that

$$e^{-(t-1)\log_2 \frac{n(t-1)}{r} - (r-(t-1))\log_2 \frac{(r-(t-1))n}{r(n-1)}}$$
$$= e^{-\frac{t-1}{\ln 2} \ln\left(\frac{n(t-1)}{r}\right) - \frac{(r-(t-1))}{\ln 2} \ln\left(\frac{(r-(t-1))n}{r(n-1)}\right)}$$
$$= \left(\frac{n(t-1)}{r}\right)^{-\frac{t-1}{\ln 2}} \left(\frac{(r-(t-1))n}{r(n-1)}\right)^{-\frac{(r-(t-1))}{\ln 2}}$$
$$= \left(\frac{r}{n(t-1)}\right)^{\frac{t-1}{\ln 2}} \left(\frac{rn-r}{rn - (t-1)n}\right)^{\frac{r-(t-1)}{\ln 2}},$$

and by denoting $r = \beta n(t-1)$, for some $\beta \geq 1$ we can rewrite the sufficient condition in (12) as follows.

$$\beta \left(1 - \frac{(\beta-1)}{\beta n - 1}\right)^{\beta n - 1} \leq (1 - R)^{\frac{\ln 2}{t-1}}. \quad (13)$$

By the definition of e, for any constant β we have that $\left(1 - \frac{(\beta-1)}{\beta n - 1}\right)^{\beta n - 1} \leq e^{-(\beta-1)}$. Hence, if $\beta e^{-\beta} \leq \frac{1}{e}(1-R)^{\frac{\ln 2}{t-1}}$ holds than (13) also holds.

By the assumption,

$$-\frac{r}{n(t-1)} e^{-\frac{r}{n(t-1)}} = -\beta e^{-\beta} \geq -\frac{1}{e}(1 - R)^{\frac{\ln 2}{t-1}}$$

or equivalently

$$\beta e^{-\beta} \leq \frac{1}{e}(1 - R)^{\frac{\ln 2}{t-1}}$$

which completes the proof. ∎

The following two claims are known results related to the Lambert W function and are given as part of the proof of Theorem 6.

Claim 7. [11, Section IV] For any real numbers $-\frac{1}{e} \leq x < 0$ and y, the equation $ye^y = x$ has exactly two solutions which are given by $y = W_0(x)$ and $y = W_{-1}(x)$, where W_0 and W_{-1} are branches of the Lambert W function.

Claim 8. [8, Theorem 1] For any $u > 0$ we have that

$$-1 - \sqrt{2u} - u < W_{-1}(-e^{-u-1}) < -1 - \sqrt{2u} - \frac{2}{3}u.$$

Proof of Theorem 6:

Denote $x = \frac{1}{e}(1-R)^{\frac{\ln 2}{t-1}}$ and $y = \frac{r}{n(t-1)}$, by Claim 7, the equation

$$-\frac{r}{n(t-1)}e^{-\frac{r}{n(t-1)}} = -ye^{-y} = -x = -\frac{1}{e}(1-R)^{\frac{\ln 2}{t-1}}$$

has exactly two solutions which are $y = -W_0(-x)$ and $y = -W_{-1}(-x)$. Note that for any $y \geq 1$ the function $-ye^{-y}$ is continuous and monotonically increasing with y. Hence, for any given $R = \frac{k}{n}$ only the branch W_{-1} is relevant. This implies that for $r \geq n(t-1)$, Equation (9) holds if and only if $y \geq -W_{-1}(-x)$.

By Claim 8, we know that $-W_{-1}(-e^{-u-1}) < 1 + \sqrt{2u} + u$ for any $u > 0$. We can rewrite $-x$ as

$$-x = -\frac{1}{e}(1-R)^{\frac{\ln 2}{t-1}} = -e^{\frac{\ln 2}{t-1}\ln(1-R)-1}$$

with $u = -\frac{\ln 2}{t-1}\ln(1-R) > 0$ to obtain

$$-W_{-1}(-x) < 1 + \sqrt{2u} + u$$
$$= 1 + \sqrt{-\frac{2\ln 2}{t-1}\ln(1-R)} - \frac{\ln 2}{t-1}\ln(1-R)$$

Hence, a sufficient condition for $\mathbb{E}[X^{(r)}] \leq n - k$ is that

$$y = \frac{r}{n(t-1)} \geq 1 + \sqrt{-\frac{2\ln 2}{t-1}\ln(1-R)} - \frac{\ln 2}{t-1}\ln(1-R),$$

or equivalently,

$$r \geq n(t-1) - n\ln 2\ln(1-R) + n(t-1)\sqrt{-\frac{2\ln 2}{t-1}\ln(1-R)}.$$
∎

APPENDIX G

Proof of Theorem 7:

We denote by ω_i the probability of collecting an error-free new strand, given that $i-1$ strands were collected. In this case, $\omega_i = \alpha\frac{n-(i-1)}{n} = \alpha\frac{n-i+1}{n}$ We denote by ω_i the probability of collecting an error-free new strand, given that $i-1$ strands were collected. In this case, $\omega_i = \alpha\frac{n-(i-1)}{n} = \alpha\frac{n-i+1}{n}$. We let t_i be the time to collect a new error-free strand, given $i-1$ such strands were already sampled. Since t_i is geometric

random variable it holds that $t_i = \frac{1}{\omega_i}$. Thus, from the linearity of expectation, we have that

$$E[\omega_\alpha(n,k)] = \mathbb{E}[t_1 + t_2 + \dots t_n] - \mathbb{E}[t_k + t_{k+1} + \dots t_n]$$
$$= \mathbb{E}[t_1] + \mathbb{E}[t_2] + \dots \mathbb{E}[t_n] - \mathbb{E}[t_k] + \mathbb{E}[t_{k+1}] + \dots \mathbb{E}[t_n]$$
$$= \frac{n}{\alpha}\left(\frac{1}{1} + \frac{1}{2} + \dots + \frac{1}{n}\right) - \frac{n}{\alpha}\left(\frac{1}{1} + \frac{1}{2} + \dots + \frac{1}{n-k}\right)$$
$$= \frac{n}{\alpha}(H_n - H_{n-k}).$$
∎

APPENDIX H

Proof of Claim 4:

Let t_i for $1 \leq i \leq \rho$ be the number of draws to collect the i-th strand in J after the $(i-1)$-th strand from J was collected. Note that $\widehat{\tau}_J(\mathcal{C}) = \sum_{i=1}^{\rho} t_i$. Additionally, observe that t_i is a geometric random variable with success probability $p_i = \frac{\rho-i+1}{n}$ and $\mathbb{E}[t_i] = \frac{1}{p_i}$. Hence, by the linearity of the expectation we have that

$$\mathbb{E}[\widehat{\tau}_J(\mathcal{C})] = \mathbb{E}\left[\sum_{i=1}^{\rho} t_i\right]$$
$$= \sum_{i=1}^{\rho} \mathbb{E}[t_i] = \sum_{i=1}^{\rho} \frac{n}{\rho-i+1}$$
$$= n\sum_{i=1}^{\rho} \frac{1}{i} = nH_\rho.$$
∎

APPENDIX I

Proof of Theorem 8:

Denote $\rho_A = |A|, \rho_B = |B|$. For a set of indices $J \subseteq [n]$, let $\lambda_J(r-1)$ be the number of different options to draw strands in the first $r-1$ draws such that for at least one of the indices $j \in J$, the strand x_j was not drawn. Additionally, let $\lambda(r-1)$ be the number of different options to draw strands in the first $r-1$ draws such that the i-th information strand cannot be retrieved from the set of drawn strands. Note that since $\mathcal{D}(i) = \{A, B\}$, we have that $\lambda(r-1)$ is the number of different options to draw strands in the first $r-1$ draws such that at least one strand from A and at least one strand from B were not drawn. Hence.

$$\lambda_{A\cup B}(r-1) = \lambda_A(r-1) + \lambda_B(r-1) - \lambda(r-1),$$

and

$$\lambda(r-1) = \lambda_A(r-1) + \lambda_B(r-1) - \lambda_{A\cup B}(r-1)$$
$$\stackrel{(a)}{=} \sum_{j=1}^{\rho_A}\binom{\rho_A}{j}(-1)^{j+1}(n-j)^{r-1}$$
$$+ \sum_{j=1}^{\rho_B}\binom{\rho_B}{j}(-1)^{j+1}(n-j)^{r-1}$$
$$- \sum_{j=1}^{\rho_A+\rho_B}\binom{\rho_A+\rho_B}{j}(-1)^{j+1}(n-j)^{r-1},$$

where (a) follows from the inclusion-exclusion principle. Using the tail sum formula for the expectation, we have that

$$\mathbb{E}\left[\tau_i(\mathcal{C})\right] = \sum_{r=1}^{\infty} \frac{\lambda(r-1)}{n^{r-1}} = \sum_{r=1}^{\infty} \sum_{j=1}^{\rho_A} \frac{\binom{\rho_A}{j}(-1)^{j+1}(n-j)^{r-1}}{n^{r-1}}$$
$$+ \sum_{r=1}^{\infty} \sum_{j=1}^{\rho_B} \frac{\binom{\rho_B}{j}(-1)^{j+1}(n-j)^{r-1}}{n^{r-1}}$$
$$- \sum_{r=1}^{\infty} \sum_{j=1}^{\rho_A+\rho_B} \frac{\binom{\rho_A+\rho_B}{j}(-1)^{j+1}(n-j)^{r-1}}{n^{r-1}}.$$

Next, we analyze the first term in the latter expression and the other two terms can be analyzed similarly.

$$\sum_{r=1}^{\infty} \sum_{j=1}^{\rho_A} \frac{\binom{\rho_A}{j}(-1)^{j+1}(n-j)^{r-1}}{n^{r-1}}$$
$$= \sum_{r=1}^{\infty} \sum_{j=1}^{\rho_A} \binom{\rho_A}{j}(-1)^{j+1}\left(1-\frac{j}{n}\right)^{r-1}$$
$$\overset{(a)}{=} \sum_{j=1}^{\rho_A} \binom{\rho_A}{j}(-1)^{j+1} \sum_{r=1}^{\infty}\left(1-\frac{j}{n}\right)^{r-1}$$
$$\overset{(b)}{=} \sum_{j=1}^{\rho_A} \binom{\rho_A}{j}(-1)^{j+1}\frac{n}{j} = n\sum_{j=1}^{\rho_A} \binom{\rho_A}{j}\frac{(-1)^{j+1}}{j}$$
$$\overset{(c)}{=} nH_{\rho_A}.$$

We note that (a) holds since the sum is absolutely convergent. (b) follows since $\left\{\left(1-\frac{j}{n}\right)^{r-1}\right\}_{r=1}^{\infty}$ is a geometric series. The equality (c) can be observed by considering Euler's integral representation of the harmonic numbers [33], $H_{\rho_A} = \int_0^1 \frac{1-x^{\rho_A}}{1-x}dx$. using the latter we have that

$$H_{\rho_A} = \int_0^1 \frac{1-x^{\rho_A}}{1-x}dx = \int_0^1 \frac{1-(1-y)^{\rho_A}}{y}dy$$
$$= \sum_{j=1}^{\rho_A}\left(\binom{\rho_A}{j}(-1)^{j+1}\int_0^1 y^{j-1}dy\right)$$
$$= \sum_{j=1}^{\rho_A} \binom{\rho_A}{j}\frac{(-1)^{j+1}}{j}.$$

Thus,

$$\mathbb{E}\left[\tau_i(\mathcal{C})\right] = n\cdot\left(H_{\rho_A} + H_{\rho_B} - H_{(\rho_A+\rho_B)}\right),$$

which concludes the proof. ∎

APPENDIX J

Proof of Theorem 9:

Let $W_{k,n}$ be the random variable that represents the number of samples needed to obtain k distinct coupons where each draw is taken from a pool of n total coupons. We denote by $W_{k,n}(x)$ the generating function for $W_{k,n}$. For $x < \frac{1}{1-\frac{n-k+1}{n}} = \frac{n}{k-1}$ it is known [29] that,

$$W_{k,n}(x) = \sum_{r=0}^{x} P[W_{k,n}=r]\cdot x^r = \prod_{i=1}^{k}\frac{(n-i+1)x}{n-(i-1)x}.$$

Additionally, let $V_{r,n}$ be the random variable that represents the number of distinct coupons in the first r draws, where each coupon is taken from a pool of n total coupons. Note that

$$P[W_{k,n}=r]$$
$$= P[V_{r-1,n}=k-1]\cdot P[V_{r,n}=k|V_{r-1,n}=k-1]$$
$$= \frac{n-(k-1)}{n}P[V_{r-1,n}=k-1],$$

and hence,

$$P[V_{r-1,n}=k-1] = \frac{n}{n-(k-1)}P[W_{k,n}=r]. \quad (14)$$

We let $D_{k,i,n}$ denote the random variable that represents the required number of draws to obtain k distinct coupons or to retrieve coupon i (whichever occurs first), where each draw is taken from a pool of n total coupons. We denote by $D_{k,i,n}(x)$ the generating function for $D_{k,i,n}$. To this end we define $D_{k,i,n}^{(j)}$ for $0 \le j \le k-1$, to be the random variable that represents the number of samples needed to obtain j distinct coupons (each not equal to the i-th coupon), followed by the i-th coupon. Additionally, let $D_{k,i,n}^{(k)}$, be the random variable that represents the number of samples needed to obtain k distinct coupons (each not equal to the i-th coupon).

It holds that,

$$P[D_{k,i,n}^{(k)}=r] = \left(1-\frac{1}{n}\right)^r\cdot P[W_{k,n-1}=r],$$

and

$$D_{k,i,n}^{(k)}(x) = \sum_{r=0}^{\infty} x^r\cdot P[D_{k,i,n}^{(k)}=r]$$
$$= \sum_{r=0}^{\infty} x^r\cdot\left(1-\frac{1}{n}\right)^r\cdot P[W_{k,n-1}=r]$$
$$= W_{k,n-1}\left(\left(1-\frac{1}{n}\right)x\right)$$
$$= \prod_{\ell=1}^{k}\frac{(n-\ell)(1-\frac{1}{n})x}{n-1-(\ell-1)(1-\frac{1}{n})x}.$$

For $1 \le j \le k-1$, using (14), we have that

$$P[D_{k,i,n}^{(j)}=r] = \left(1-\frac{1}{n}\right)^{r-1}\cdot\left(\frac{1}{n}\right)\cdot P[V_{r-1,n-1}=j]$$
$$= \left(1-\frac{1}{n}\right)^{r-1}\cdot\left(\frac{1}{n}\right)\cdot\frac{n-1}{n-1-j}\cdot P[W_{j+1,n-1}=r]$$

and,

$$D_{k,i,n}^{(j)}(x) = \sum_{r=0}^{\infty} x^r\cdot P[D_{k,i,n}^{(j)}=r]$$
$$= \sum_{r=0}^{\infty} x^r\cdot\left(1-\frac{1}{n}\right)^{r-1}\cdot\left(\frac{1}{n}\right)\cdot\frac{n-1}{n-1-j}\cdot P[W_{j+1,n-1}=r]$$
$$= \frac{n-1}{n-1-j}\cdot\frac{1}{n}\left(1-\frac{1}{n}\right)^{-1}\sum_{r=0}^{\infty} x^r\left(1-\frac{1}{n}\right)^r P[W_{j+1,n-1}=r]$$
$$= \frac{1}{n-1-j}\cdot W_{j+1,n-1}\left(\left(1-\frac{1}{n}\right)x\right)$$
$$= \frac{1}{n-1-j}\prod_{\ell=1}^{j+1}\frac{(n-\ell)\left(1-\frac{1}{n}\right)x}{n-1-(\ell-1)\left(1-\frac{1}{n}\right)x}.$$

Note that, $P[D_{k,i,n}^{(0)} = r] = \frac{1}{n}$ if $r = 1$ and otherwise $P[D_{k,i,n}^{(0)} = r] = 0$. Therefore, we have that $D_{k,i,n}^{(0)}(x) = \frac{x}{n}$. Next, we present $D_{k,i,n}(x)$ as a function of $D_{k,i,n}^{(j)}$ for $0 \leq j \leq k$.

$$\begin{aligned} D_{k,i,n}(x) &= \sum_{r=0}^{\infty} x^r \cdot P[D_{k,i,n} = r] \\ &= \sum_{r=0}^{\infty} x^r \sum_{j=0}^{k} P[D_{k,i,n}^{(j)} = r] \\ &= \sum_{j=0}^{k} \sum_{r=0}^{\infty} x^r \cdot P[D_{k,i,n}^{(j)} = r] \\ &= \sum_{j=0}^{k} D_{k,i,n}^{(j)}(x). \end{aligned}$$

From the above, it can be derived that,

$$\begin{aligned} \mathbb{E}[D_{k,i,n}] &= D'_{k,i,n}(1) \\ &= \frac{1}{n} + \sum_{j=1}^{k-1} \frac{n(n-(j+1))(\psi(-n) - \psi(j+1-n))}{n(n-(j+1))} \\ &\quad + \frac{n(n-k)(\psi(-n) - \psi(k-n))}{n} \\ &= \frac{1}{n} + \sum_{j=1}^{k-1} (\psi(-n) - \psi(j+1-n)) + (n-k)(\psi(-n) - \psi(k-n)), \end{aligned}$$

where $\psi(z) \triangleq \int_0^{\infty} \left(\frac{e^{-t}}{t} - \frac{e^{-zt}}{1-e^{-t}} \right) dt$ is the digamma function. In [18], it is claimed that for any $z \in \mathbb{C}$ and $j \in \mathbb{N}$, we have that, $\psi(z+j) = \psi(z) + \sum_{h=1}^{j} \frac{1}{z+h-1}$, which implies that, $\psi(-n) - \psi(j-n) = H_n - H_{n-j}$.

Hence,

$$\begin{aligned} \mathbb{E}[\tau_i(\mathcal{C})] &= \mathbb{E}[D_{k,i,n}] = \frac{1}{n} + \sum_{j=1}^{k-1} (H_n - H_{n-j-1}) \\ &\quad + (n-k)(H_n - H_{n-k}) \\ &= \frac{1}{n} + (n-1)H_n - (n-k)H_{n-k} - \sum_{j=1}^{k-1} H_{n-j-1} \\ &= \frac{1}{n} + (n-1)H_n - (n-k)H_{n-k} \\ &\quad - \left(\sum_{j=1}^{n-2} H_j - \sum_{j=1}^{n-k-1} H_j \right) \\ &\overset{(a)}{=} \frac{1}{n} + (n-1)H_n - (n-k)H_{n-k} \\ &\quad - ((n-1)(H_{n-1} - 1 - (n-k)(H_{n-k} - 1))) \\ &= \frac{1}{n} + (n-1)(H_n - H_{n-1} + 1) - (n-k) \\ &= \frac{1}{n} + \frac{n-1}{n} + (n-1) - n + k \\ &= k, \end{aligned}$$

where (a) follows since for any integer $n > 0$ we have that $\sum_{j=1}^{n} H_j = (n+1)H_n - n$.

APPENDIX K

Proof of Theorem 10:

Let $\mathcal{C}_k$ be the systematic $(2k, k)$ code that is defined by $\mathbf{U}_k = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k)$ and

$$\mathbf{X}_k = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k, \boldsymbol{u}_1 + \boldsymbol{u}_2, \ldots, \boldsymbol{u}_{k-1} + \boldsymbol{u}_k, \boldsymbol{u}_k + \boldsymbol{u}_1).$$

Similarly to Example 2, we have that

$$P[\tau_1(\mathcal{C}) \geq r] = \sum_{i=1}^{2k-3} P[\tau_1(\mathcal{C}) \geq r | \mathcal{E}_{r-1} = i] \cdot P[\mathcal{E}_{r-1} = i],$$

and

$$P[\mathcal{E}_r = i] = \frac{\binom{2k}{i}}{(2k)^r} \sum_{j=0}^{i} \binom{i}{j} (-1)^{i-j} (i-j)^r.$$

Since $P[\tau_1(\mathcal{C}) \geq r | \mathcal{E}_{r-1} = i]$ does not depend on r, let us denote $P_i = P[\tau_1(\mathcal{C}) \geq r | \mathcal{E}_{r-1} = i]$. We have that

$$\begin{aligned} \mathbb{E}[\tau_1(\mathcal{C})] &= \sum_{r=0}^{\infty} P[\tau_1(\mathcal{C}) \geq r] \\ &= 1 + \sum_{r=1}^{\infty} \sum_{i=1}^{2k-3} P_i \cdot P[\mathcal{E}_{r-1} = i] \\ &= 1 + \sum_{i=1}^{2k-3} P_i \sum_{r=1}^{\infty} P[\mathcal{E}_{r-1} = i] \\ &= 1 + \sum_{i=1}^{2k-3} P_i \sum_{r=1}^{\infty} \binom{2k}{i} \sum_{j=0}^{i} (-1)^{i-j} \binom{i}{j} \left(\frac{j}{2k} \right)^{r-1} \\ &= 1 + \sum_{i=1}^{2k-3} P_i \binom{2k}{i} \sum_{j=0}^{i} (-1)^{i-j} \binom{i}{j} \sum_{r=0}^{\infty} \left(\frac{j}{2k} \right)^r \\ &= 1 + \sum_{i=1}^{2k-3} P_i \binom{2k}{i} (-1)^i \sum_{j=0}^{i} (-1)^j \binom{i}{j} \cdot \frac{2k}{2k-j} \\ &= 1 + \sum_{i=1}^{2k-3} P_i \cdot 2k \cdot \binom{2k}{i} \cdot (-1)^i \cdot \frac{(-1)^i}{(2k-i)\binom{2k}{i}} \\ &= 1 + \sum_{i=1}^{2k-3} P_i \cdot \frac{2k}{(2k-i)}. \end{aligned}$$

Hence, our goal is to calculate P_i. Let $A(2k-1, i)$ be the number of options to draw $r-1$ strands such that $\boldsymbol{u}_1$ cannot be recovered from this set of draws, knowing that the set of different encoded strands that were drawn is of size exactly i. Then, we have that $P_i = \frac{A(2k-1,i)}{\binom{2k}{i}}$.

To present a recursive expression for the values $A(2k-1, i)$, we describe an equivalent way to represent the options that contribute to $A(2k - 1, i)$. Let $G_k = (V, E)$ be the directed graph with the $2k - 1$ nodes that correspond to the symbols in $\mathcal{X}$ excluding $\boldsymbol{u}_1$. The set E consists of the following edges:

- For each $2 \leq j \leq k - 1$, the vertex $\boldsymbol{u}_j + \boldsymbol{u}_{j+1}$ has four outgoing edges. Two green outgoing edges to the nodes $\boldsymbol{u}_j$ and $\boldsymbol{u}_{j-1} + \boldsymbol{u}_j$, and two blue outgoing edges to the nodes $\boldsymbol{u}_{j+1}$ and $\boldsymbol{u}_{j+1} + \boldsymbol{u}_{j+2}$ (where $\boldsymbol{u}_{j+2} = \boldsymbol{u}_1$ if $j = k - 1$).
- There are two blue outgoing edges from $\boldsymbol{u}_1 + \boldsymbol{u}_2$, to the nodes $\boldsymbol{u}_2$ and $\boldsymbol{u}_2 + \boldsymbol{u}_3$.

- There are two green outgoing edges from $\boldsymbol{u}_k + \boldsymbol{u}_0$, to the nodes $\boldsymbol{u}_k$ and $\boldsymbol{u}_{k-1} + \boldsymbol{u}_k$.

Denote the nodes $\boldsymbol{u}_1 + \boldsymbol{u}_2$ and $\boldsymbol{u}_k + \boldsymbol{u}_1$ by S_2 and S_k, respectively. Additionally, denote the nodes $\boldsymbol{u}_j$, for $2 \leq j \leq k$ by *ending nodes*. For a set $J \subseteq [2k]\backslash\{1\}$, let $G_k^{(J)}$ be the subgraph of G_k that contains all the nodes that correspond to J (considering their locations in $\mathbf{X}$). Note that any set $J \subseteq [2k]$ of size i is not a retrieval set of $\boldsymbol{u}_1$ if and only if the subgraph of $G_k^{(J)}$, does not contain a monochromatic path from S_2 or S_k to one of the ending nodes (if S_2, S_k is not in $G_k^{(J)}$, we say that there is no such path from S_2, S_k, respectively). Hence, $A(2k-1, i)$ is equal to the number of subgraphs $G_k^{(J)}$ of G_k, such that $J \subseteq [2k]\backslash\{1\}$ and $G_k^{(J)}$ does not contain a monochromatic path from S_2 or S_k to one of the ending nodes. Denote the nodes of $G_k^{(J)}$ by V' and consider the following cases.

1) If $S_2, S_k \notin V'$ then any such a subgraph $G_k^{(J)}$ cannot contain a valid monochromatic path and there are $\binom{2k-3}{i}$ such subgraphs.
2) If $S_2 \in V'$ then we have that $\boldsymbol{u}_2 \notin V'$ and there are $A(2k-3, i-1)$ such sub-graphs.
3) If $S_k \in V'$ then we have that $\boldsymbol{u}_k \notin V'$ and there are $A(2k-3, i-1)$ such sub-graphs.
4) If $S_2, S_k \in V'$ then we have that $\boldsymbol{u}_2, \boldsymbol{u}_k \notin V'$ and there are $A(2k-5, i-2)$ such subgraphs.

Thus,

$$A(2k-1, i) = \binom{2k-3}{i} + 2A(2k-3, i-1) - A(2k-5, i-2).$$

By denoting $B(k, i) = A(2k+1, i)$, we can write the latter as

$$B(k, i) = \binom{2k-1}{i} + 2B(k-1, i-1) - B(k-2, i-2),$$

for any $k \geq 2$, $i \geq 2$, and for all $k \geq 0$ we have that $B(k, 0) = 1$ and $B(k, 1) = 2k+1$. Additionally $A(1, 2) = 1$, for $i \geq 2$ we have $A(0, i) = 0$ and for $i \geq 3$ we have $B(1, i) = 0$.

Thus, we have that

$$\mathbb{E}[\tau_1(\mathcal{C})] = 1 + \sum_{i=1}^{2k-3} B(k, i) \cdot \frac{2k}{(2k-i)\binom{2k}{i}}.$$

■

APPENDIX L

Proof of Theorem 11:

For any $r \geq 1$ draws, let us denote by ε_i^r the random variable that governs the number of strands drawn from $\mathbf{X}_s$

(from the r draws), where s is an integer and $u_i \in \mathbf{U}_s$. Then we have that

$$\mathbb{E}[\tau_i(\mathcal{C}^\gamma)] = \sum_{r=1}^{\infty} P[\tau_i(\mathcal{C}^\gamma) \geq r]$$

$$= \sum_{r=1}^{\infty} \sum_{z=0}^{\infty} P[\varepsilon_i^{r-1} = z] \cdot P\left[\tau_i(\mathcal{C}^\gamma) \geq r | \varepsilon_i^{r-1} = z\right]$$

$$\overset{(a)}{=} \sum_{r=1}^{\infty} \sum_{z=0}^{r-1} P[\varepsilon_i^{r-1} = z] \cdot P\left[\tau_i(\mathcal{C}^\gamma) \geq r | \varepsilon_i^{r-1} = z\right]$$

$$= \sum_{r=1}^{\infty} \sum_{z=0}^{r-1} \binom{r-1}{z}\left(\frac{1}{\gamma}\right)^z\left(1 - \frac{1}{\gamma}\right)^{r-z-1} P\left[\tau_i(\mathcal{C}^\gamma) \geq r | \varepsilon_i^{r-1} = z\right]$$

$$= \sum_{r=1}^{\infty} \sum_{z=0}^{r-1} \binom{r-1}{z}\left(\frac{1}{\gamma}\right)^z\left(1 - \frac{1}{\gamma}\right)^{r-z-1} \cdot P\left[\tau_i(\mathcal{C}) \geq z + 1\right]$$

$$= \sum_{z=0}^{\infty} P\left[\tau_i(\mathcal{C}) \geq z + 1\right] \sum_{r=z+1}^{\infty} \binom{r-1}{z}\left(\frac{1}{\gamma}\right)^z\left(1 - \frac{1}{\gamma}\right)^{r-z-1}$$

$$= \sum_{z=0}^{\infty} P\left[\tau_i(\mathcal{C}) \geq z + 1\right] \sum_{r=z}^{\infty} \binom{r}{z}\left(\frac{1}{\gamma}\right)^z\left(1 - \frac{1}{\gamma}\right)^{r-z}$$

$$\overset{(b)}{=} \sum_{z=0}^{\infty} P\left[\tau_i(\mathcal{C}) \geq z + 1\right] \cdot \gamma$$

$$= \sum_{z=1}^{\infty} P\left[\tau_i(\mathcal{C}) \geq z\right] \cdot \gamma = \gamma\mathbb{E}\left[\tau_i(\mathcal{C})\right],$$

where equality (a) follows from the fact that the probability to collect $z > r - 1$ unique strands from $\mathbf{X}_s$, using only $r-1$ draws is zero for any integer s, i.e., $P[\varepsilon_i^{r-1} = z] = 0$. To see that equality (b) holds, recall that $\sum_{r=0}^{\infty} x^r = \frac{1}{1-x}$, and by taking the derivative of the latter z times we get

$$\sum_{r=z}^{\infty} r \cdot (r-1) \cdots (r-z+1)x^{r-z} = \frac{z!}{(1-x)^{z+1}},$$

which is equivalent to

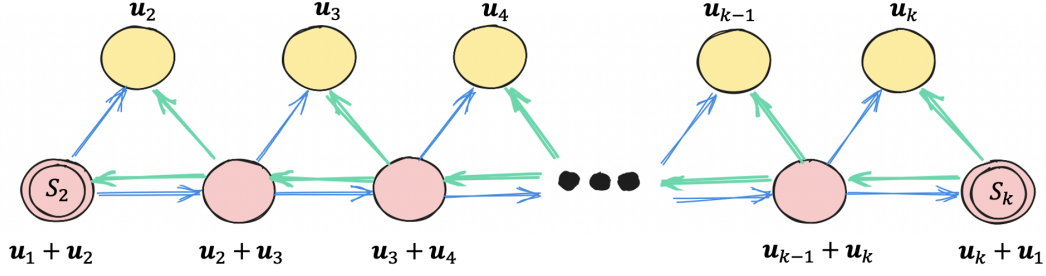$$\sum_{r=z}^{\infty} \binom{r}{z} x^{r-z} = \frac{1}{(1-x)^{z+1}}.$$

Lastly, by substituting $x = 1 - \frac{1}{\gamma}$, equality (a) follows. ■

APPENDIX M

Proof of Theorem 12:

We prove the claim only for $k = 2$, while the proof for $k = 3$ relies on the exact same ideas. Assume w.l.o.g. that we want to retrieve $\boldsymbol{u}_1$. For simplicity of the analysis, also assume that $\mathbf{X}$ contains two information stands $\boldsymbol{u}_1, \boldsymbol{u}_2$ (without multiplicity), and that each of them can be drawn with probability $\frac{p}{2}$. First note that since $(\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{(1-p)n})$ belongs to a $[(1-p)n+2, 2]$ MDS code, any two distinct strands form a retrieval set for $\boldsymbol{u}_1$, and hence the only case in which we didn't retrieve $\boldsymbol{u}_1$ in r draws, is when we draw the same strand (which is not $\boldsymbol{u}_1$) r times.

- $\tau_1(\mathcal{C}_{n,p}^2) = 1$ only in case we draw $\boldsymbol{u}_1$ in the first draw which happens with probability $\frac{p}{2}$.

Fig. 6: Schematic description of G_k

- $\tau_1(\mathcal{C}_{n,p}^2) = r$ for $r \geq 2$ only if the first $r - 1$ draws are of the strand $\boldsymbol{x} \neq \boldsymbol{u}_1$ and the last draw is of a different strand. Hence we have that

$$P[\tau_1(\mathcal{C}_{n,p}^2) = r]$$
$$= \left(\frac{p}{2}\right)^{r-1}\left(1 - \frac{p}{2}\right) + (1-p)n \cdot \left(\frac{1}{n}\right)^{r-1}\left(1 - \frac{1}{n}\right).$$

Thus,

$$\mathbb{E}[\tau_1(\mathcal{C}_{n,p}^2)] = \sum_{r=1}^{\infty} P[\tau_i(\mathcal{C}_{n,p}) = r] \cdot r$$

$$= \frac{p}{2} + \sum_{r=2}^{\infty} r\left(\left(\frac{p}{2}\right)^{r-1}\left(1 - \frac{p}{2}\right) + (1-p)n\left(\frac{1}{n}\right)^{r-1}\left(1 - \frac{1}{n}\right)\right)$$

$$= \frac{p}{2} + \left(1 - \frac{p}{2}\right)\sum_{r=2}^{\infty} r\left(\frac{p}{2}\right)^{r-1} + (1-p)\left(1 - \frac{1}{n}\right)\sum_{r=2}^{\infty} r\left(\frac{1}{n}\right)^{r-2}.$$

For n large enough $\left(1 - \frac{1}{n}\right)\sum_{r=2}^{\infty} r\left(\frac{1}{n}\right)^{r-2} \approx 2$ and hence for n large enough we have that

$$\mathbb{E}[\tau_1(\mathcal{C}_{n,p}^2)] \approx \frac{p}{2} + \left(1 - \frac{p}{2}\right)\sum_{r=2}^{\infty} r\left(\frac{p}{2}\right)^{r-1} + 2(1-p)$$

$$= \frac{p}{2} + \left(1 - \frac{p}{2}\right)\frac{p(4-p)}{(2-p)^2} + 2(1-p)$$

$$= \frac{p}{2} + \frac{p(4-p)}{2(2-p)} + 2(1-p).$$

This expression is minimized when $p = 2 - \sqrt{2}$ and in this case we have

$$\frac{p}{2} + \frac{p(4-p)}{2(2-p)} + 2(1-p) \approx 1.83.$$

Note that even though the optimal p is irrational, since the latter function is continuous for $p \in (0,1)$, we can get as close as we want to this optimum value. Hence, we have that

$$\mathbb{E}[\tau_1(\mathcal{C}_{n,p}^2)] \approx 1.83 = 0.9143k.$$

■

APPENDIX N

Proof of Corollary 7:

From Theorem 13, for any (n,k) code $\mathcal{C}$, we have that

$$T_{\max}^{\mathcal{C}} \geq n - \frac{n(n-k)}{k} \cdot (H_n - H_{n-k}).$$

Thus, we have that

$$\lim_{i \to \infty} \frac{T_{\max}^{\mathcal{C}_i}}{k_i} \geq \lim_{i \to \infty} \frac{1}{k_i}\left(n_i - \frac{n_i(n_i - k_i)}{k_i}(H_{n_i} - H_{n_i - k_i})\right)$$

$$= \lim_{i \to \infty} \frac{n_i}{k_i}\left(1 - \frac{n_i - k_i}{k_i} \cdot (H_{n_i} - H_{n_i - k_i})\right)$$

$$= \lim_{i \to \infty} \frac{1}{R} - \frac{(1-R)}{R^2}(H_{n_i} - H_{n_i - k_i})$$

$$= \frac{1}{R} - \frac{1-R}{R^2}\log\left(\frac{1}{1-R}\right).$$

It can be verified that in this case if R approaches zero, one, then the latter expression approaches $\frac{1}{2}, 1$, respectively. ■

REFERENCES

[1] H. Abraham, R. Gabrys, and E. Yaakobi, "Covering All Bases: The Next Inning in DNA Sequencing Efficiency," in *IEEE International Symposium on Information Theory (ISIT)*, 2024.

[2] 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.

[3] J.L. Banal, T.R. Shepherd, J. Berleant, H. Huang, M. Reyes, C.M. Ackerman, P.C. Blainey, and M. Bathe, "Random access DNA memory using Boolean search in an archival file storage system," *Nature Materials*, vol. 20, pp. 1272–1280, 2021.

[4] D. Bar-Lev, O. Sabary, R. Gabrys and E. Yaakobi, "Cover your bases: How to minimize the sequencing coverage in DNA storage systems," IEEE International Symposium on Information Theory (ISIT), Taipei, Taiwan, pp. 370–375, 2023.

[5] 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.

[6] H.P.J. Buermans and J.T. den Dunnen, "Next generation sequencing technology: Advances and applications," *Biochimica et Biophysica Acta (BBA)-Molecular Basis of Disease*, vol. 1842, no. 10 pp. 1932–1941, 2014.

[7] S. Chandak, K. Tatwawadi, B. Lau, J. Mardia, M. Kubit, J. Neu, P. Griffin, M. Wootters, T. Weissman, H. Ji, "Improved read/write cost tradeoff in DNA-based data storage using LDPC codes," *Annual Allerton Conference on Communication, Control, and Computing*, 2019.

[8] 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.

[9] 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.

[10] I. Csiszar, "ℓ-divergence geometry of probability distributions and minimization problems," *The Annal of Probabilty*, vol. 3, no. 1, pp. 146–158, 1975.

[11] 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.

[12] 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.

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

[14] A. Fazeli, A. Vardy, and E. Yaakobi, "Codes for distributed PIR with optimal storage overhead," *Proc. IEEE Int'l Symp. on Information Theory*, pp. 2852–2856, Hong Kong, Jun. 2015.

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

[16] 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.

[17] A. Gruica, D. Bar-Lev, A. Ravagnani, and E. Yaakobi, "Reducing coverage depth in DNA storage: A combinatorial perspective on random access efficiency," *IEEE International Symposium on Information Theory (ISIT)*, 2024.

[18] 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.

[19] R. Heckel, G. Mikutis, and R. N. Grass, "A Characterization of the DNA data storage channel," *Scientific Reports*, vol. 9, no. 9663, 2019.

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

[21] 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.

[22] B. Lau, S. Chandak, S. Roy, K. Tatwawadi, M. Wootters, T. Weissman, and H.P. Ji, "Magnetic DNA random access memory with nanopore readouts and exponentially-scaled combinatorial addressing," *BiorXiv*, 10.1101/2021.09.15.460571, 2021.

[23] 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.

[24] E. M. LeProust, B. J. Peck, K. Spirin, H. B. McCuen, B. Moore, E. Namsaraev, M. H. Caruthers, "Synthesis of high-quality libraries of long (150mer) oligonucleotides by a novel depurination controlled process," *Nucleic Acids Research*, no. 38, pp. 2522–2540, 2010.

[25] D. J. Newman, "The double dixie cup problem," *The American Mathematical Monthly*, vol. 67, no. 1, pp. 58-61, 1960.

[26] 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.

[27] S.J. Park, S. Kim, J. Jeong, A. No, J.S. No, H. Park, "Reducing cost in DNA-based data storage by sequence analysis-aided soft information decoding of variable-length reads," *Bioinformatics*, vol. 39, no. 9, 2023.

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

[29] 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.

[30] I. Preuss, B. Galili, Z. Yakhini, and L. Anavy, "Sequencing coverage analysis for combinatorial DNA-based storage systems," *IEEE Transactions on Molecular, Biological, and Multi-scale Communications*, 2024.

[31] I. Preuss, M. Rosenberg, Z. Yakhini, and L. Anavy, "Efficient DNA-based data storage using shortmer combinatorial encoding," *Scientific Reports*, vol. 14, no. 1, p. 7731, 202

[32] J. Rydning. "Worldwide IDC Global DataSphere Forecast, 2022–2026: Enterprise Organizations Driving Most of the Data Growth," *International Data Corporation (IDC)*, 2022.

[33] C. E. Sandifer, How Euler Did It. Washington, DC: Mathematical Association of America, 2007.

[34] J. Sima, N. Raviv, and J. Bruck, "On coding over sliced information," *IEEE International Symposium on Information Theory (ISIT)*, pp. 767–771, 2019.

[35] 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.

[36] R. Sokolovskii, P. Agarwal, L. A. Croquevielle, Z. Zhou, and T. Heinis, "Coding Over Coupon Collector Channels for Combinatorial Motif-Based DNA Storage," *arXiv preprint*, 2024.

[37] A. Vardy and E. Yaakobi, "Private information retrieval without storage overhead: Coding instead of replication," to appear *IEEE Journal on Selected Areas in Inform. Theory*, 2023.

[38] 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.

[39] C. Winston, L. Organick, D. Ward, L. Ceze, K. Strauss, and Y.-J. Chen, "Combinatorial PCR method for efficient, selective oligo retrieval from complex oligo pools", *ACS Synth. Biol.*, vol. 11, pp. 1727–1734, 2022.

[40] 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.

[41] 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

[42] S.M.H.T. Yazdi, Y. Yuan, J. Ma, H. Zhao, and O. Milenkovic, "A rewritable, Random-access DNA-based storage system," *Scientific Reports*, vol. 5, pp. 1–10, 2015.

[43] 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.

Daniella Bar-Lev (Student Member, IEEE) received a B.Sc. degree in Computer Science and a B.Sc. degree in Mathematics in 2019, an M.Sc. degree in Computer Science in 2021, and a Ph.D. degree in Computer Science in 2024, all from the Technion — Israel Institute of Technology, Haifa, Israel. She is currently a Post-Doctoral Fellow in the Faculty of Computer Science at the Technion. She is a recipient of the Eric and Wendy Schmidt Postdoctoral Award for Women in Mathematical and Computing Sciences, the Blavatnik Prize for Outstanding Israeli Ph.D. Students in Computer Science, the Gutwirth Excellence Scholarship, and the Student Research Prize for Cross-PI Collaboration in Data Science of VATAT. Her research interests include algorithms, discrete mathematics, coding theory, and DNA storage.

Omer Sabary (Student Member, IEEE) received the M.Sc. degree from the Computer Science Department, Technion—Israel Institute of Technology, in 2020. He is currently pursuing a Ph.D. degree in the Henry and Marilyn Taub Faculty of Computer Science, Technion — Israel Institute of Technology. His advisors are Prof. Eitan Yaakobi and Prof. Antonia Wachter-Zeh. His research interests include coding techniques and algorithms for DNA storage systems. He was a recipient of the VATAT Excellence Award in data mining in 2020, 2023, and 2024. He also received the Jacobs-Qualcomm Excellence Fellowship for PhD Students in 2023/24

Ryan Gabrys (Member, IEEE) received the B.S. degree in mathematics and computer science from the University of Illinois at Urbana–Champaign in 2005 and the Ph.D. degree in electrical engineering from the University of California, Los Angeles, in 2014. He is currently a Scientist jointly affiliated with the Naval Information Warfare Center, California Institute for Telecommunications and Information Technology (Calit2), University of California, San Diego. His research interests broadly lie in the areas of theoretical computer science and electrical engineering, including coding theory, combinatorics, and communication theory.

Eitan Yaakobi (Senior Member, IEEE)is an Associate Professor at the Computer Science Department at the Technion — Israel Institute of Technology. He also holds a courtesy appointment in the Technion's Electrical and Computer Engineering (ECE) Department. He received the B.A. degrees in computer science and mathematics, and the M.Sc. degree in computer science from the Technion — Israel Institute of Technology, Haifa, Israel, in 2005 and 2007, respectively, and the Ph.D. degree in electrical engineering from the University of California, San Diego, in 2011. Between 2011-2013, he was a postdoctoral researcher in the department of Electrical Engineering at the California Institute of Technology and at the Center for Memory and Recording Research at the University of California, San Diego. His research interests include information and coding theory with applications to non-volatile memories, associative memories, DNA storage, data storage and retrieval, and private information retrieval. He received the Marconi Society Young Scholar in 2009 and the Intel Ph.D. Fellowship in 2010-2011. Between 2020 and 2023, he served as an Associate Editor for Coding snd Decoding for the IEEE TRANSACTIONS ON INFORMATION THEORY. Since 2016, he is affiliated with the Center for Memory and Recording Research at the University of California, San Diego, and between 2018–2022, he was affiliated with the Institute of Advanced Studies, Technical University of Munich, where he held a four-year Hans Fischer Fellowship, funded by the German Excellence Initiative and the EU 7th Framework Program. Between August 2023 and January 2024, he was a Visiting Associate Professor at the School of Physical and Mathematical Sciences at Nanyang Technological University. He is a recipient of several grants, including the ERC Consolidator Grant and the EIC Pathfinder Challenge.