Error-Correcting Codes for Combinatorial Composite DNA

Omer Sabary[*†], **Inbal Preuss**[*‡], **Ryan Gabrys**[§], **Zohar Yakhini**[†‡], **Leon Anavy**[‡], and **Eitan Yaakobi**[†]

[†]Faculty of Computer Science, Technion—Israel Institute of Technology, Haifa, Israel

[‡]Faculty of Computer Science, Reichman University, Hezliya, Israel

[§]Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA, USA

Emails: {omersabary, yaakobi}@cs.technion.ac.il, {inbalpreuss, leon.anavy}@gmail.com, rgabrys@eng.ucsd.edu, zohar.yakhini@runi.ac.il

***Abstract*—Data storage in DNA is developing as a possible solution for archival digital data. Recently, to further increase the potential capacity of DNA-based data storage systems, the combinatorial composite DNA synthesis method was suggested. This approach extends the DNA alphabet by harnessing short DNA fragment reagents, known as *shortmers*. The shortmers are building blocks of the alphabet symbols, each consisting of a fixed number of shortmers. Thus, when information is read, it is possible that one of the shortmers that forms part of the composition of a symbol is missing and therefore the symbol cannot be determined. In this paper, we model this type of error as a type of asymmetric error and propose code constructions that can correct such errors in this setup. We also provide a lower bound on the redundancy of such error-correcting codes and give an explicit encoder and decoder for our construction. Our suggested error model is also supported by an analysis of data from actual experiments that produced DNA according to the combinatorial scheme. Lastly, we also provide a statistical evaluation of the probability of observing such error events, as a function of read depth.**

I. Introduction

In the last decade, there has been notable progress in DNA storage systems, where the stability and density of DNA molecules are utilized to create robust and high-capacity data storage platforms [2], [4], [8], [19]. In standard DNA storage systems, binary data is encoded into sequences over the DNA alphabet $\{A, C, G, T\}$, in which each symbol represents a DNA base (also known as nucleotide). Then, based on these sequences, DNA molecules called *strands* are generated by a biological process termed *DNA synthesis*, that can only generate multiple copies per strand. The synthesized strands are stored in a storage container. To read back the binary information the strands are read back into their digital representation using *DNA sequencing*. The sequencing data is called *reads*, and the reads are used as an input to the decoder that retrieves the stored information. The synthesis, storage, and sequencing are error-prone processes, and thus, to retrieve the information error-correcting codes should be considered.

Recently, Anavy et al, and Choi et al. [1], [6] suggested an innovative way to extend the DNA alphabet by harnessing the inherent redundancy (multiple copies) of the synthesis and sequencing process with the use of *composite DNA symbols*. A composite DNA symbol is a representation of a position in a sequence in which there is not just a single base, but a mixture of the four DNA bases. By using a mixture of bases, the alphabet is extended with more symbols that are defined by the bases in the mixture and their ratios.

Later, Preuss et al. [12] suggested an extension to the composite DNA synthesis, which is referred to as the *combinatorial composite synthesis method* and the design of codes suitable for this emergent method is the main focus of this paper. In the combinatorial composite synthesis method, the building blocks of the composite symbols are the so-called *shortmers*. A shortmer (also known as a *motif*) is a fixed-length sequence that consists of DNA bases. The shortmers are synthesized using a standard DNA synthesis technology, and then they are connected to each other using biochemical process called *ligation* [12]. In this case, each valid composite symbol is in fact a set of $w \in \mathbb{N}^+$ distinct shortmers. Thus, the alphabet consists of sets of shortmers. To improve the data reliability, and to allow easier detection of the shortmers, they are selected as a subset of all shortmers of a specific length. Other extensions of this method can be found in [14], [20].

The alphabet symbols of the combinatorial composite method are sets of shortmers. Therefore, it is possible that one or more shortmers are not represented in the sequencing reads. In such cases, the observed set of shortmers is missing a subset of them causing an error in reading the data. In this paper, we model these cases as *asymmetric errors* and study error-correcting codes (ECCs) for the combinatorial composite synthesis. We provide constructions for such codes, including an explicit encoder, and present bounds on their redundancy.

The rest of this paper is organized as follows. Section II gives the preliminaries and defines the main problem discussed in the paper. In Section III we present a construction for a composite asymmetric error-correcting code. In Section IV we give a sphere-packing bound on such codes. In Section V explicit encoder and decoder for our construction are provided, while Section VI presents an analysis of data from previous experiments as well as an evaluation of the probabilities of our discussed error models. Lastly, in Section VII we give a lower bound on the redundancy of error-correcting codes for a more general error model. Due to lack of space, the proofs can be found in the longer version of this paper [15].

II. Definitions and Problem Statement

In the following, we represent our data as a sequence of length m where each element in the sequence is a set of shortmers. Under this representation, each set of shortmers will be represented as a binary vector of length n with exactly

[*] These authors contributed equally to this work. This project has received funding by the European Union (DiDAX, 101115134). Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. This work was also supported in part by NSF Grant CCF2212437.

979-8-3503-8284-6/24/$31.00 ©2024 IEEE

w ones in it where the location of the ones in the vector indicate which shortmers are contained in every set. Since every element in our sequence is itself a set, we will find it useful to represent our data as a set of $m \times n$ binary arrays with each row in the array specifying a set of shortmers.

A. Notations

For a positive integer n, let $[n] \triangleq \{0, \dots, n-1\}$. For a binary vector $\boldsymbol{x}$, $w_H(\boldsymbol{x})$ denotes the Hamming weight (shortly the weight) of $\boldsymbol{x}$, which is the number of ones in $\boldsymbol{x}$.

Let $\Sigma = \{A, C, G, T\}$ be the DNA alphabet and let $\ell \in \mathbb{N}$ be the shortmer length. We let $\mathcal{S} = \{\boldsymbol{s}_0, \boldsymbol{s}_1, \dots, \boldsymbol{s}_{n-1}\}$ be a set of $n > 1$ *shortmers*, $\boldsymbol{s}_i \in \Sigma^\ell$ for $i \in [n]$, which are indexed lexicographically. For $w < n$, we define the *w-combinatorial composite alphabet of $\mathcal{S}$* as $\Sigma_w^{\mathcal{S}} \triangleq \{\boldsymbol{x}_0^{\mathcal{S}}, \boldsymbol{x}_1^{\mathcal{S}}, \dots, \boldsymbol{x}_{\binom{n}{w}-1}^{\mathcal{S}}\}$, where each *combinatorial composite symbol* $\boldsymbol{x}_i^{\mathcal{S}}$, for $i \in [\binom{n}{w}]$ is a *set* of w different shortmers chosen from the shortmers set $\mathcal{S}$. For simplicity, the set of symbols in $\Sigma_w^{\mathcal{S}}$ can be abstracted as length-n binary vectors of weight w in which every bit indicates whether a shortmer in $\mathcal{S}$ belongs to the set. Thus, every $\boldsymbol{x}_i^{\mathcal{S}} \in \Sigma_w^n$ is mapped to its indicator binary vector, denoted by $\boldsymbol{x}_i \in \{0,1\}^n$ and note that $\sum_{j=0}^{n-1} x_{i,j} = w$. From this point, we refer to the composite symbols in our alphabet by their binary vector representation and denote the set of length-n binary vectors of weight w by Σ_w^n.

Example 1. In [12], the authors used the following parameters $\ell = 3$, $n = 16$ and $w = 5$,

$$\mathcal{S} = \left\{ \begin{matrix} \boldsymbol{s}_0 = AAT, \boldsymbol{s}_1 = ACA, \boldsymbol{s}_2 = ATG, \boldsymbol{s}_3 = AGC, \\ \boldsymbol{s}_4 = TAA, \boldsymbol{s}_5 = TCT, \boldsymbol{s}_6 = TTC, \boldsymbol{s}_7 = TGG, \\ \boldsymbol{s}_8 = GAG, \boldsymbol{s}_9 = GCC, \boldsymbol{s}_{10} = GTT, \boldsymbol{s}_{11} = GGA, \\ \boldsymbol{s}_{12} = CAC, \boldsymbol{s}_{13} = CCG, \boldsymbol{s}_{14} = CTA, \boldsymbol{s}_{15} = CGT \end{matrix} \right\}.$$

The set $\mathcal{S}$ was selected as a code with Hamming distance of $d = 2$. In this setup, an example of the 99-th composite symbol of the alphabet is $\boldsymbol{x}_{99}^{\mathcal{S}} = (1,1,0,1,0,0,1,1,0,0,0,0,0,0,0,0)$, which represents the set consisting of the shortmers $\{\boldsymbol{s}_0, \boldsymbol{s}_1, \boldsymbol{s}_3, \boldsymbol{s}_6, \boldsymbol{s}_7\}$.

A sequence of length m over a composite alphabet $\Sigma_w^{\mathcal{S}}$ is denoted by $\mathcal{X}^{\mathcal{S}} = (\boldsymbol{x}_{i_0}^{\mathcal{S}}, \dots, \boldsymbol{x}_{i_{m-1}}^{\mathcal{S}}) \in (\Sigma_w^{\mathcal{S}})^m$. This sequence can be abstracted as an $m \times n$ binary matrix $\mathcal{X}$, in which each row is matched with its corresponding composite symbol from $\Sigma_w^{\mathcal{S}}$. That is,

$$\mathcal{X} = \begin{pmatrix} \boldsymbol{x}_{i_0} \\ \vdots \\ \boldsymbol{x}_{i_{m-1}} \end{pmatrix} = \begin{pmatrix} x_{i_0,0}, x_{i_0,1}, x_{i_0,2} \dots, x_{i_0,n-1} \\ \vdots \\ x_{i_{m-1},0}, x_{i_{m-1},1}, \dots, x_{i_{m-1},n-1} \end{pmatrix},$$

and note that for any $h \in [m]$, $\sum_{j=0}^{n-1} x_{i_h,j} = w$.

In this paper, we consider *the combinatorial composite-DNA channel*, which receives an $m \times n$ matrix $\mathcal{X}$, and outputs a noisy version of $\mathcal{X}$, denoted by $\mathcal{Y}$. Similarly, we denote the rows of the matrix $\mathcal{Y}$ by $\boldsymbol{y}_h$, $h \in [m]$, such that $\boldsymbol{y}_h$ is a noisy version of $\boldsymbol{x}_{i_h}$

$$\mathcal{Y} = \begin{pmatrix} \boldsymbol{y}_0 \\ \vdots \\ \boldsymbol{y}_{m-1} \end{pmatrix} = \begin{pmatrix} y_{0,0}, y_{0,1}, \dots, y_{0,n-1} \\ \vdots \\ y_{m-1,0}, y_{m-1,1}, \dots, y_{m-1,n-1} \end{pmatrix}.$$

Lastly, since the exact shortmers in the set $\mathcal{S}$ do not matter but only the number of shortmers, we refer to the combinatorial composite alphabet from now on by the set Σ_w^n and a length-m sequence is simply an $m \times n$ matrix in $\Sigma_w^{m \times n}$, where $\Sigma_w^{m \times n}$ refers to the set of all $m \times n$ matrices in which the weight of every row is w.

B. Problem Statement

Next, we define *composite-asymmetric errors*, which is our main interest in this paper.

Definition 1. Composite asymmetric errors. For a positive integer e and a row vector $\boldsymbol{x}_i = (x_0, \dots, x_{n-1}) \in \Sigma_w^n$, we say that the corresponding channel output $\boldsymbol{y}_i = (y_0, \dots, y_{n-1}) \in \Sigma_{w-e}^n$, suffers from e *composite-asymmetric errors* if $y_i \leq x_i$, and $\sum_{i=0}^{n-1} y_i = w - e$.

Definition 1 can be extended to matrices as described below.

Definition 2. (t,e)-composite asymmetric errors. For positive integers e and t and a matrix $\mathcal{X} = (\boldsymbol{x}_0, \dots, \boldsymbol{x}_{m-1})^T \in \Sigma_w^{m \times n}$, we say that the channel output matrix $\mathcal{Y} = (\boldsymbol{y}_0, \dots, \boldsymbol{y}_{m-1})^T$, suffers from (t,e)-*composite asymmetric errors* if at most t rows of $\mathcal{X}$ are noisy, each of them suffers from at most e composite-asymmetric errors.

A *length-m (n,w)-composite code* $\mathcal{C}$ is a set of matrices over $\Sigma_w^{m \times n}$ and every codeword in $\mathcal{C}$ is referred to as a *composite codeword*. We say that a length-m (n,w)-composite code is a (t,e)-*composite-asymmetric ECC (in short (t,e)-CAECC)*, if it can correct any (t,e)-composite asymmetric error. Such a code will be referred as an $[m,(n,w);t,e]$-*composite code*.

We denote by $A(m,n,w)$ the size of the set of all binary matrices of dimension $m \times n$, in which each row is of weight exactly w, that is $A(m,n,w) = |\Sigma_w^{m \times n}| = \binom{n}{w}^m$. We denote by $A(m,n,w;t,e)$ the size of the largest $[m,(n,w);t,e]$-composite code. For a composite code $\mathcal{C} \subseteq \Sigma_w^{m \times n}$, we define its redundancy to be $r(\mathcal{C}) \triangleq \log(|A(m,n,w)|) - \log(|\mathcal{C}|)$. Furthermore, we denote by $r(m,n,w;t,e)$ the minimum redundancy of such a composite code.

The main goal of this paper is to study (t,e)-CAECCs and more specifically to solve the following problem.

Problem 1. Find the value of $A(m,n,w;t,e)$, the size of the largest $[m,(n,w);t,e]$-composite code and correspondingly find the minimum redundancy $r(n,w,m;t,e)$.

Although the problem of coding for the asymmetric channel has been studied extensively in the past, our setup departs from previous works such as [3], [9], [10], [17] in that the sequences over which our codes are being developed over satisfy a local weight constraint. In particular, recall that each row of our codeword matrices has exactly w ones in it, and this extra information can be leveraged to dramatically reduce the redundancy of our resulting coding schemes. To the best of our knowledge this setup has not been studied before, and one of the goals in this work will be to identify parameter regimes where we can design efficient codes capable of correcting such errors that are larger than traditional asymmetric error-correcting codes.

III. Code Constructions

In this section we give a code construction for (t,e)-CAECCs. For any length-n binary vector $\boldsymbol{x} =$

$(x_0, x_1, \ldots, x_{n-1}) \in \{0,1\}^n$, positive integers ℓ and p, we define the *ℓ-VT-syndrome over p* [17] of $\boldsymbol{x}$, denoted by $\boldsymbol{s}_\ell^p(\boldsymbol{x})$, as follows $\boldsymbol{s}_\ell^p(\boldsymbol{x}) \triangleq (\sum_{i=0}^{n-1} i^\ell x_i) \bmod p$. Note that the VT-syndrome is usually defined such that $\boldsymbol{s}_\ell^{n+1}(\boldsymbol{x}) = (\sum_{i=0}^{n-1} i^\ell x_i) \bmod (n+1)$ and therefore $\boldsymbol{s}_\ell^{n+1}(\boldsymbol{x}) \in [n+1]$, however, we use the above definition with a prime number to correct multiple errors and to be able to construct outer codes using tensor product codes [18]. Therefore, according to our definition, we have that $\boldsymbol{s}_\ell^p(\boldsymbol{x}) \in \mathbb{F}_p$, and for the rest of the paper, it is assumed that p is the smallest prime such that $p \geq n$ and according to Bertrand's Postulate [5], we assume that $n \leq p \leq 2n$.

We also define the *e-complete-VT-syndrome over p*, denoted by $\mathbf{S}_e^p(\boldsymbol{x})$, to be $\mathbf{S}_e^p(\boldsymbol{x}) \triangleq (\boldsymbol{s}_1^p(\boldsymbol{x}), \boldsymbol{s}_2^p(\boldsymbol{x}), \ldots, \boldsymbol{s}_e^p(\boldsymbol{x}))$, and note that $\mathbf{S}_e^p(\boldsymbol{x})$ can be interpreted as an element in $\mathbb{F}_{p^e}$. We extend this definition to matrices $\mathcal{X} \in \Sigma_w^{m \times n}$, whose rows are given by $\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{m-1}$, and define the *$e$-complete-VT-syndrome-vector over p* of $\mathcal{X}$, denoted by $\mathbf{S}_e^p(\mathcal{X})$, to be the vector in which its i-th entry corresponds to the e-complete-VT-syndrome of the i-th row in $\mathcal{X}$. That is,

$$\mathbf{S}_e^p(\mathcal{X}) \triangleq (\mathbf{S}_e^p(\boldsymbol{x}_0), \mathbf{S}_e^p(\boldsymbol{x}_1), \ldots, \mathbf{S}_e^p(\boldsymbol{x}_{m-1})) \in (\mathbb{F}_{p^e})^m.$$

Next, a construction of an $[m,(n,w);t,e]$-composite code is presented.

Construction 1. Let $e \geq 1$, p be the smallest prime number such that $p \geq n$, and $\mathcal{C}_t$ be an $[m,k,t+1]$ code over $\mathbb{F}_{p^e}$ capable of correcting t erasures. If $m \leq p^e$, then $\mathcal{C}_t$ is selected as an MDS code with $k = m - t$. The code $\mathcal{C}_{m,n,w}^{(t,e)}$ is defined as follows. $\mathcal{C}_{m,n,w}^{(t,e)} = \{\mathcal{X} \in \Sigma_w^{m \times n} : \mathbf{S}_e^p(\mathcal{X}) \in \mathcal{C}_t\}$.

Theorem 1. The code $\mathcal{C}_{m,n,w}^{(t,e)}$ is a (t,e)-CAECC.

Proof. Let $\mathcal{Y}$ be an $m \times n$ matrix, obtained from a composite codeword $\mathcal{X}$ in the code $\mathcal{C}_{m,n,w}^{(t,e)}$. We denote by $1 \leq i_1, \ldots, i_t \leq m$, the t indices of the rows of $\mathcal{Y}$ that suffer from e composite-asymmetric errors (if any of these rows suffer from less errors, the proof can be adapted accordingly). That is, $w_H(\boldsymbol{y}_{i_1}) = w_H(\boldsymbol{y}_{i_2}) = \cdots = w_H(\boldsymbol{y}_{i_t}) = w - e$. We prove that it is possible to decode the codeword by proving that any of the above rows can be uniquely decoded. Without the loss of generality, we show the latter for $\boldsymbol{y}_{i_1}$, and the same proof works for any of the other erroneous rows. As $\mathcal{C}_t$ is capable of correcting t erasures, by using the decoder of the code $\mathcal{C}_t$, it is possible to decode the correct e-complete-VT-syndrome of the i_1-th row, $\mathbf{S}_e^p(\boldsymbol{x}_{i_1})$, and thus also the ℓ-VT-syndrome, $\boldsymbol{s}_\ell^p(\boldsymbol{x}_{i_1})$, for $1 \leq \ell \leq e$. We let $h_1 < \cdots < h_e$ be the set of e indices corresponding to the locations in which $\boldsymbol{y}_{i_1}$ had asymmetric errors. Then we have that for any $1 \leq \ell \leq e$,

$$h_1^\ell + \cdots + h_e^\ell \equiv (\boldsymbol{s}_\ell^p(\boldsymbol{x}_{i_1}) - \boldsymbol{s}_\ell^p(\boldsymbol{y}_{i_1})) \bmod p.$$

Hence, we get e equations per erroneous row. In Theorem 1 in [7], it was shown that these equations have an equivalent polynomial form in which the roots are the indices h_ℓ, $1 \leq \ell \leq e$. This polynomial form can be obtained by considering the Newton-Girard formulas [16]. Thus, as was done in [7], Vieta's formula can be used to get the set of roots of the polynomial, which is known to be unique by Lagrange theorem [11]. This proves the theorem. □

Construction 1 leads to the following corollary.

Corollary 1. For $m \leq p^e$, it holds that $r(n,w,m;t,e) \leq et\log(p) \leq et\log(2n)$, and for $e = 1$, we have that, $r(m,n,w;t,1) \leq t\ \log(n)$.

IV. Bounds on the Size of Composite Asymmetric Error-Correcting Codes

This section provides a sphere packing bound on the size of $[m,(n,w);t,e]$ composite code. We first note that our code $\mathcal{C}$ is defined over the space $\Sigma_w^{m \times n}$. Hence, given a codeword $\mathcal{X} \in \mathcal{C}$, any asymmetric error changes the weight of at least one of $\mathcal{X}$'s rows. Thus, all resulting matrices do not necessarily have the same structure, and therefore the sphere-packing bound cannot be used directly. However, in this section we show how by defining a specific distance and proving some properties on CAECCs, it is indeed possible to get a sphere packing bound for a (t,e)-CAECC for any t and e.

First, we define the Hamming distance between two vectors $\boldsymbol{x}, \boldsymbol{y}$ of the same length, denoted by $d_H(\boldsymbol{x},\boldsymbol{y})$ as the number of positions in which their bits are different. Next, we define the e-Hamming distance of two matrices $\mathcal{X}, \mathcal{Y} \in \Sigma_w^{m \times n}$.

Definition 3. Let $\mathcal{X}, \mathcal{Y} \in \Sigma_w^{m \times n}$, and integer $e \geq 0$. Then,

$$d_{e\text{-}H}(\mathcal{X},\mathcal{Y}) \triangleq \begin{cases} \infty, \text{if } \exists i \in [n] : d_H(\boldsymbol{x}_i, \boldsymbol{y}_i) > e, \\ |\{i : \boldsymbol{x}_i \neq \boldsymbol{y}_i\}|, \text{otherwise.} \end{cases}$$

The *e-Hamming distance* of a code $\mathcal{C}$, denoted by $d_{e\text{-}H}(\mathcal{C})$, is defined as the minimum e-Hamming distance between any two different codewords in $\mathcal{C}$. That is, $d_{e\text{-}H}(\mathcal{C}) \triangleq \min_{\mathcal{X},\mathcal{Y} \in \mathcal{C}, \mathcal{X} \neq \mathcal{Y}} \{d_{e\text{-}H}(\mathcal{X},\mathcal{Y})\}$. Finally, we define the *e-Hamming error ball of radius t* of a word $\mathcal{X} \in \Sigma_w^{m \times n}$, denoted by $B_{e\text{-}H}(\mathcal{X},t)$, as the set of all words in $\Sigma_w^{m \times n}$ that have e-Hamming distance of t or less from $\mathcal{X}$

$$B_{e\text{-}H}(\mathcal{X},t) = \{\mathcal{Y} \in \Sigma_w^{m \times n} : d_{e\text{-}H}(\mathcal{X},\mathcal{Y}) \leq t\}.$$

Lemma 1. For any two integers $t \leq m, e \leq w$, it holds that a code $\mathcal{C} \subseteq \Sigma_w^{m \times n}$ is a (t,e)-CAECC if and only if $d_{2e\text{-}H}(\mathcal{C}) \geq t+1$.

Lemma 1 states that a code $\mathcal{C} \subseteq \Sigma_w^{m \times n}$ is a (t,e)-CAECC if and only if its $2e$-Hamming distance is at least $t+1$. It is further possible to show that the e-Hamming distance is a metric for any $e \geq 0$. The above implies that the radius-$\lfloor \frac{t}{2} \rfloor$ $2e$-Hamming error balls of $\mathcal{C}$ are mutually disjoint. Based on this observation, we can compute an explicit sphere-packing bound on (t,e)-CAECCs.

Theorem 2. It holds that,

$$A(m,n,w;t,e) \leq \frac{\binom{n}{w}^m}{\left(\frac{m}{\lfloor \frac{t}{2} \rfloor}\right)^{\lfloor \frac{t}{2} \rfloor} \cdot \left(\frac{4 \cdot w(n-w)}{e^2}\right)^{\frac{e}{2} \cdot \lfloor \frac{t}{2} \rfloor}}, \text{ and}$$

$$\begin{aligned} r(m,n,w;t,e) \geq & \left\lfloor \frac{t}{2} \right\rfloor \log(m) - \left\lfloor \frac{t}{2} \right\rfloor \log\left(\left\lfloor \frac{t}{2} \right\rfloor\right) \\ & + \frac{e}{2}\left\lfloor \frac{t}{2} \right\rfloor \log(w(n-w)) + 2\frac{e}{2}\left\lfloor \frac{t}{2} \right\rfloor - e\left\lfloor \frac{t}{2} \right\rfloor \log(e). \end{aligned}$$

Lastly, we have another bound on $A(m,n,w;t,e)$.

Theorem 3. It holds that, $A(m,n,w;t,e) \leq \frac{\binom{n}{w}^m}{\binom{n-w+e}{e}^t}$, and $r(m,n,w;t,e) \geq t\log\binom{n-w+e}{e}$.

Note that the last bound does not depend on m, and thus, it will be effective only for smaller values of m, while for larger values, it is preferable to apply the bound from Theorem 2.

To give some intuition to our results, let us assume the simplified case in which $m = n = p$ is a prime number, and $w = \frac{n+1}{2}$. In this case, Construction 1 and Corollary 1 show that there exists, for example a $(2,2)$-CAECC with redundancy which is at most $4\log(n)$, while Theorem 2 and Theorem 3 show that the redundancy is at least $\log(n) + \log(n^2-1) - 2$ and $2\log(n^2+n+3) - 6$, respectively. Thus, our construction is only constant away from optimality.

V. Explicit Encoder and Decoder

Next, we describe explicit encoder and decoder for the code $\mathcal{C}_{m,n,w}^{(t,1)}$ when $n = p$ is a prime number, and $m \leq n$. The overall redundancy of the code is $t\log(n)$. In the next algorithm, we show how to encode $m \cdot \lfloor \log\binom{n}{w} \rfloor - t\log(n)$ information bits into a codeword in $\mathcal{C}_{m,n,w}^{(t,1)}$. Recall that under these given parameters, the code $\mathcal{C}_{m,n,w}^{(t,1)}$ is defined as all the matrices $\mathcal{X} \in \Sigma_w^{m\times n}$, such that $\mathbf{S}_1^n(\mathcal{X}) \in \mathcal{C}_t$, where $\mathcal{C}_t$ is an $[m, m-t]$ MDS code over $\mathbb{F}_n$.

Before moving forward into the description of the encoding algorithm, we define the (i,n,w) *coset of the VT-syndrome over n*, denoted by $\boldsymbol{s}_1^n(i,n,w)$, as the set of all vectors $\boldsymbol{x} \in \Sigma_w^n$, such that $\boldsymbol{s}_1^n(\boldsymbol{x}) \equiv i \mod n$. Theorem 4 states that for coprime w and n, the cosets $\boldsymbol{s}_1^n(i,n,w)$ for all $i \in \mathbb{F}_n$ have the same size.

Theorem 4. Let $n \in \mathbb{N}^+$ and let $w < n$ such that w and n are coprime, we have that for all $i \in [n]$, $|\boldsymbol{s}_1^n(i,n,w)| = \frac{\binom{n}{w}}{n}$.

Our encoder assumes a mapping function that receives $\lfloor \log\binom{n}{w} \rfloor$ bits and encodes them into vectors over Σ_w^n. This mapping is denoted by E_1. Furthermore, from Theorem 4, we have that since the size of all the cosets $\boldsymbol{s}_1^n(i,n,w)$ is the same, it is possible to create a mapping that receives a pair of a syndrome in $\mathbb{F}_n$ and $\log(\lfloor \frac{\binom{n}{w}}{n} \rfloor)$ bits of information and encodes them into a vector $\boldsymbol{x} \in \Sigma_w^n$. We denote this mapping by E_2. Our suggested encoder is described in Algorithm 1.

Algorithm 1 Encoding Algorithm

```
procedure ENCODER
    Input. The encoder receives $m \cdot \lfloor \log\binom{n}{w} \rfloor - t\log(n)$ bits and encodes them into a codeword in $\mathcal{C}_{w,n,m}^{(t,1)}$. For this purpose the all zero binary matrix $\mathcal{X}$ of dimension $m \times n$ is initialized.
    Step 1. Take the first $(m-t) \cdot \lfloor \log\binom{n}{w} \rfloor$ bits, and encode them using $E_1$ into $(m-t)$ combintorial-composite symbols over $\Sigma_w^n$. Fill the resulted binary vectors in the first $(m-t)$ rows of $\mathcal{X}$.
    Step II. Compute the phantom-syndrome vector of the first $m-t$ rows of $\mathcal{X}$, $(\boldsymbol{s}_1^n(\boldsymbol{x}_1), \ldots, \boldsymbol{s}_1^n(\boldsymbol{x}_{m-t}))$.
    Step III. Encode the phantom-syndrome vector $(\boldsymbol{s}_1^n(\boldsymbol{x}_1), \ldots, \boldsymbol{s}_1^n(\boldsymbol{x}_{m-t}))$ using the encoder of the $[m, m-t]$ MDS code $\mathcal{C}_t$. By the end of this step, we obtained the encoded phantom syndrome vector, $\boldsymbol{s}_1^n(\mathcal{X}) = (\boldsymbol{s}_1^n(\boldsymbol{x}_1), \ldots, \boldsymbol{s}_1^n(\boldsymbol{x}_{m-t}), r_1, \ldots, r_t))$. The symbols $r_1, \ldots, r_t$ are the redundancy symbols over $[n]$ of the code $\mathcal{C}_t$. The redundancy symbols $r_1, \ldots, r_t$ can be interpreted as syndromes of the last $t$ rows of the matrix, i.e., rows $m-t+1, \ldots, m$. In particular, it holds that $\boldsymbol{s}_1^n(\mathcal{X}) = (\boldsymbol{s}_1^n(\boldsymbol{x}_1), \ldots, \boldsymbol{s}_1^n(\boldsymbol{x}_{m-t}), r_1, \ldots, r_t) = (\boldsymbol{s}_1^n(\boldsymbol{x}_1), \ldots, \boldsymbol{s}_1^n(\boldsymbol{x}_{m-t}), \boldsymbol{s}_1^n(\boldsymbol{x}_{m-t+1}), \ldots, \boldsymbol{s}_1^n(\boldsymbol{x}_m))$.
    Step IV.. The last $t$ rows of the matrix $\mathcal{X}$ are encoded as follows. For $1 \leq i \leq t$, the $m-t+i$-th row of $\mathcal{X}$ is encoded with $E_2$ by considering the combination of $\boldsymbol{s}_1^n(\boldsymbol{x}_{m-t+i}) = r_i$ and $\log(\lfloor \frac{\binom{n}{w}}{n} \rfloor)$ bits of information.
    Output. The matrix $\mathcal{X}$ is returned as output.
```

VI. Simulation and Statistics

In this section, we analyze data from previous experiments [12], [20] to support our channel model and error characterization. Furthermore, we provide an evaluation of the error probabilities of observing asymmetric errors.

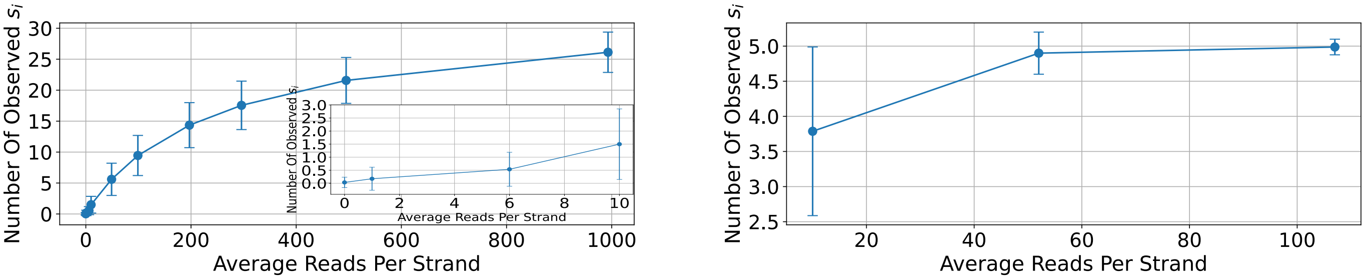
A. Statistics on real data

To emphasize the importance of error correction code in recovering data effectively, Fig. 1 shows the direct correlation that exists between the quality and quantity of sequencing reads and the number of shortmers observed. An insufficient number of observed shortmers can lead to errors in data recovery, which are defined in this paper as asymmetric errors. Fig. 1 depicts data from two combinatorial composite shortmers experiments with different synthesis protocols and different sequencing technologies [12] [20]. The plots represent sampling of reads from the overall full set of reads with varying sampling rates. The number of observed unique shortmers is plotted against the average number of reads per strand. Fig. 1a shows results from [20] where a single combinatorial synthesis cycle was demonstrated (meaning $m = 1$) with a $n = 96$ and $w = 32$. The sequencing in this experiment was performed using Oxford Nanopore MinION. Clearly, even with an average coverage of 1,000 reads we could not recover all 32 shortmers. Fig. 1b shows the results from [12] where four combinatorial synthesis cycles ($m = 4$) were demonstrated with $n = 16$ and $w = 5$. The sequencing was performed using Illumina MiSeq. In this case, a coverage of 100 reads was sufficient for recovering all five shortmers. Note that it is possible that more than five shortmers can be observed due to wrong classification or other experimental errors.

B. Evaluation of error probability

Fig. 2 and Table I depict the probabilities of observing composite asymmetric errors directly calculated using the coupon collector's model described in [13]. It should be noted that this calculation is based on the combinatorial factor w alone, ignoring n, as the model assumes uniform sampling of the w observed s_i. In Fig. 2 the probability of observing e errors or more is shown as a function of the number of analyzed reads (R) for a single combinatorial letter ($m = 1$) using combinatorial factor $w = 5$. As expected, the error probability decreases as more reads are analyzed. However, it is likely to observe several composite asymmetric errors when analyzing 10-20 reads. This emphasizes the need for efficient ECCs for correcting composite asymmetric errors.

Table I shows direct calculation of the probability of a message encoded using the (t,e) code to be successfully decoded. That is, the table presents the probability of observing at most t letters with at most e errors in each for a combinatorial word of



(a) Data of [20] with $n = 96$, $w = 32$, $m = 1$, 8 different strands.

(b) Data of [12] with $n = 16$, $w = 5$, $m = 4$, 2 different strands.

Fig. 1: Asymmetric combinatorial errors in experimental results. The x-axis represents the average reads per strand, in sampling from actual NGS data. The y-axis shows the number of observed s_i. Midpoints represent the mean count of observed s_i, and the whiskers represent the std of 10 repeated samplings aggregated over the different strands to each experiment.

length $m = 10$ and combinatorial factor $w = 5$. For instance, the probability of observing at most one symbol with at most one error is presented in the first cell ($p(1, 1) = 0.0137$). Clearly, the probability increases as t and e increase.

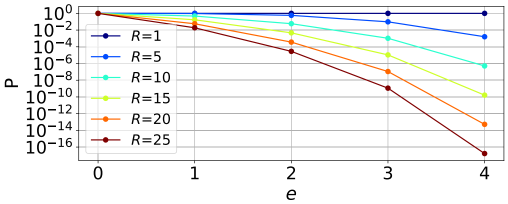


Fig. 2: Probability to observe e asymmetric errors or more in a single combinatorial symbols. The x-axis indicates e or more errors, each line represents a different number of analyzed reads (R) and the y-axis shows the error probability. Results for $w = 5, R = 1, 5, 10, 20, 25, e = 0, 1, \ldots, 4$.

e\t	1	2	3	4	5	6	7	8	9	10
1	0.3138	0.5967	0.8031	0.9019	0.9343	0.9417	0.9429	0.9430	0.9430	0.9430
2	0.3201	0.6187	0.8425	0.9526	0.9897	0.9984	0.9998	1.0000	1.0000	1.0000
3	0.3201	0.6187	0.8425	0.9526	0.9898	0.9984	0.9998	1.0000	1.0000	1.0000
4	0.3201	0.6187	0.8425	0.9526	0.9898	0.9984	0.9998	1.0000	1.0000	1.0000

TABLE I: Probability of observing at most (t, e) asymmetric combinatorial errors. Rows correspond to t. Columns correspond to e. In each calculation $w = 4, R = 10, m = 10, e = 1, 2, \ldots, 5, t = 1, 2, \ldots, 10$ and $n > w$.

VII. Extensions of the Asymmetric Error Model

This section discusses more generalized error models that include asymmetric errors.

A. (t_1, t_2)-CAECCs

In practice, as can be seen in our analysis of data from previous experiments in Section VI, asymmetric errors of more than $e = 2$ shortmers rarely happen. However, it is more likely that t_1 rows can suffer from 1 asymmetric errors each, while $t_2 < t_1$ rows suffer from $e = 2$ asymmetric errors. Codes that can correct errors of this pattern are termed $(t_1, t_2)-$CAECC. For two integers $t_1 \geq t_2$, we assume $\mathcal{C}_{t_1+t_2}, \mathcal{C}_{t_2}$ are codes correcting $t_1 + t_2, t_2$ erasures (respectively) over $\mathbb{F}_p$, where p is the smallest prime number, such that $p \geq n$. Construction 1 can be extended to form a $(t_1, t_2)-$CAECC as follows.

Construction 2. Let p be the smallest prime number such that $n \leq p$ and $m \leq p$. Then, we have that,

$$\mathcal{C}_{(t_1,t_2)} = \{\mathcal{X} \in \Sigma_w^{m\times n} : \mathbf{S}_1^p(\mathcal{X}) \in \mathcal{C}_{t_1+t_2}, \mathbf{S}_2^p(\mathcal{X}) \in \mathcal{C}_{t_2}\}.$$

Using the same techniques that were used in the proof of Construction 1 it can be shown that the code $\mathcal{C}_{(t_1,t_2)}$ can correct up to t_1 rows with 1 asymmetric error and t_2 rows with 2 asymmetric errors.

Corollary 2. There exists a (t_1, t_2)-CAECC with redundancy of $(t_1 + 2t_2)\log(p)$.

The latter construction can be further extended to correct e_1 in t_1 rows and $e_2 \geq e_1$ errors in t_2 rows.

Construction 3. Let $e_2 \geq e_1 \geq 1$, and let p be the smallest prime number such that $n \leq p$. Then, we have that,

$$\mathcal{C}_{(t_1,e_1,t_2,e_2)} = \{\mathcal{X} \in \Sigma_w^{m\times n} : \mathbf{S}_i^p(\mathcal{X}) \in \mathcal{C}_{t_1+t_2}, \mathbf{S}_j^p(\mathcal{X}) \in \mathcal{C}_{t_2},$$
$$\text{for any } 1 \leq i \leq e_1 \text{ and } 1 \leq j \leq e_2\}.$$

Similarly to Theorem 3, we can show that a lower bound on the redundancy of the code from the latter construction is $t_1\log(\binom{n-w+e_1}{e_1}) + t_2\log(\binom{n-w+e_2}{e_2})$.

B. 2-CAECC

Lastly, we study the case in which the channel can introduce up to $e = 2$ composite asymmetric errors *anywhere* in the codeword $\mathcal{X} \in \Sigma_w^{m\times n}$. Such codes are termed 2-*composite asymmetric error-correcting codes (*2*-CAECC)*. Next, we give a sufficient condition for a code to correct $e = 2$ asymmetric errors. For a word $\mathcal{X} \in \Sigma_w^{m\times n}$ we define $A_e(\mathcal{X})$ as the e-*asymmetric error ball* of $\mathcal{X}$ as all the words that can be obtained from $\mathcal{X}$ by introducing up to two asymmetric errors.

Lemma 2. Let $\mathcal{X}, \mathcal{Y} \in \Sigma_w^{m\times n}$ we have that if $A_2(\mathcal{X}) \cap A_2(\mathcal{Y}) \neq \emptyset$ then, $B_{2\text{-}H}(\mathcal{X}, 1) \cap B_{2\text{-}H}(\mathcal{Y}, 1) \neq \emptyset$. Hence, given a code $\mathcal{C} \subseteq \Sigma_w^{m\times n}$, if $\mathcal{C}$ satisfies $B_{2\text{-}H}(\mathcal{X}, 1) \cap B_{2\text{-}H}(\mathcal{Y}, 1) = \emptyset$ for any $\mathcal{X}, \mathcal{Y} \in \mathcal{C}$ then $\mathcal{C}$ is 2-CAECC.

Lemma 2 implies that by considering the size of the e-Hamming error balls of radius 1, it is possible to apply a sphere packing bound on 2-CAECCs. For this purpose, we denote by $A(m, n, w, e)$ the maximum size of an e-CAECC, where $r(m, n, w, e)$ denotes its minimum redundancy.

Theorem 5. It holds that, $A(m, n, w, e) \leq \frac{\binom{n}{w}}{mw(n-w)}$, and $r(m, n, w, e) \geq \log(m \cdot (w) \cdot (n - w))$.

Lastly, it should be noted that using Construction 2 it is possible to create a a 2-CAECC that by Corollary 2 has a redundancy of $3\log(p)$. Assuming $m = n = p$ prime and $w = \frac{p+1}{2}$, we get by Theorem 5 that an optimal redundancy of such code is $\log(p\frac{p+1}{2}\frac{p-1}{2}) = 3\log(p) + \log(1 - \frac{1}{p^2}) - 2 \geq 3\log(p) - 2.5$, which implies that Construction 2 is only a constant away from optimality.

References

[1] L. Anavy, I. Vaknin, O. Atar, R. Amit, and Z. Yakhini, “Data storage in DNA with fewer synthesis cycles using composite DNA letters" *Nature Biotechnology*, vol. 37, no. 10, pp. 1229–1236, 2019.

[2] D. Bar-Lev, I. Orr, O. Sabary, T. Etzion, and E. Yaakobi, “Deep DNA storage: Scalable and robust DNA storage via coding theory and deep learning," *arXiv preprint arXiv:2109.00031*, 2021.

[3] B. Bose and S. Al-Bassam, “On Systematic Single Asymmetric Error Correcting Codes,” *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 669-672, Mar. 2000.

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

[5] J. Bertrand, “Mémoire sur le nombre de valeurs que peut prendre une fonction quand on y permute les lettres qu’elle renferme,", *Journal de l’École Royale Polytechnique (in French)*, vol. 18, no. 30, pp. 123-140, 1845.

[6] Y. Choi, T. Ryu, A. C. Lee, H. Choi, H. Lee, J. Park, S. Song, S. Kim, H. Kim, W. Park, and S. Kwon, “High information capacity DNA-based data storage with augmented encoding characters using degenerate bases," *Scientific Reports*, vol. 9, no. 6582, 2019.

[7] L. Dolecek, “Towards longer lifetime of emerging memory technologies using number theory," *IEEE Globecom Workshops*, Miami, FL, USA, pp. 1936-1940, 2010.

[8] N. Goldman, P. Bertone, S. Chen, C. Dessimoz, E. M. LeProust, B. Sipos, and E. Birney, “Towards practical, high-capacity, low-maintenance information storage in synthesized DNA," *Nature*, vol. 494, no. 7435, pp. 77–80, 2013.

[9] M. Grassl, P. Shor, G. Smith, J. Smolin and B. Zeng, “New Constructions of Codes for Asymmetric Channels Via Concatenation,” *IEEE Transactions on Information Theory*, vol. 61, no. 4, pp. 1879-1886, Apr. 2015.

[10] T. Kløve, “Error correcting codes for the asymmetric channel,” Technical Report, Dept. of Informatics, University of Bergen, 1981. (Updated in 1995.)

[11] M. B. Nathanson, *Additive Number Theory: The Classical Bases*, Springer, 2010.

[12] I. Preuss, R. Michael, Z. Yakhini, and L. Anavy, “Efficient DNA-based data storage using shortmer combinatorial encoding" *Scientific Reports* vol. 14, no. 7731, 2024.

[13] I. Preuss, G. Ben, Z. Yakhini, and L. Anavy, “Sequencing coverage analysis for combinatorial DNA-based storage systems" *bioRxiv* 2024.01.10.574966, doi: https://doi.org/10.1101/2024.01.10.574966, 2024.

[14] N. Roquet, S. Bhatia, S. Flickinger, S. Mihm, M. Norsworthy, D. Leake, and H. Park, "DNA-based data storage via combinatorial assembly," *bioRxiv*, https://www.biorxiv.org/content/10.1101/2021.04.20.440194v1, 2021.

[15] O. Sabary, I. Preuss, R. Gabrys, Z. Yakhini, L. Anavy, and E. Yaakobi, “Error-correcting codes for combinatorial DNA composite," *arxiv*, 2401.15666, http://arxiv.org/abs/2401.15666, 2024.

[16] R. Seroul and D. O’Shea, *Programming for Mathematicians*, Springer, 2000.

[17] R. R. Varshamov, and G. M. Tenenholtz, “A code for correcting a single asymmetric error," *Automatica i Telemekhanika*, vol. 26, no. 2, pp. 288–292, 1965.

[18] J. K. Wolf, “An introduction to tensor product codes and applications to digital storage systems," In *IEEE Information Theory Workshop (ITW)*, pp. 6–10, 2006.

[19] S. H. T. Yazdi, R. Gabrys, and O. Milenkovic, “Portable and error-free DNA-based data storage," *Scientific Reports*, vol. 7, no. 5011, 2017.

[20] Y. Yiqing, N. Pinnamaneni, S. Chalapati, C. Crosbie, and R. Appuswamy. “Scaling Logical Density of DNA storage with Enzymatically-Ligated Composite Motifs" *Scientific Reports* vol. 13 2023. doi: https://doi.org/10.1038/s41598-023-43172-0.