Insertion and Deletion Correction in Polymer-based Data Storage

Anisha Banerjee[1], Antonia Wachter-Zeh[2], and Eitan Yaakobi[3]

[1,2]Institute for Communications Engineering, Technical University of Munich (TUM), Munich, Germany
[3]Department of Computer Science, Technion - Israel Institute of Technology, Haifa 3200003, Israel
Email: [1]anisha.banerjee@tum.de, [2]antonia.wachter-zeh@tum.de, [3]yaakobi@cs.technion.ac.il

***Abstract*—Synthetic polymer-based storage promises to accommodate the ever-increasing demand for archival storage. It involves designing molecules of distinct masses to represent the respective bits $\{0, 1\}$, followed by the synthesis of a polymer of molecular units that reflects the order of bits in the information string. The stored data can be read by means of a tandem mass spectrometer, that fragments the polymer into shorter substrings and provides their corresponding masses, from which the *composition*, i.e., the number of 1s and 0s in the concerned substring can be inferred. Prior works tackled the problem of unique string reconstruction from the set of all possible compositions, called the *composition multiset*. This was accomplished either by determining which string lengths always allow unique reconstruction, or by formulating coding constraints to facilitate the same for all string lengths. Additionally, error-correcting schemes to deal with substitution errors caused by imprecise fragmentation during the readout process, have also been suggested. This work extends previously considered error models that were mainly confined to substitutions of compositions. Our new error models consider insertions and deletions of compositions. The robustness of the reconstruction codebook proposed by Pattabiraman *et al.* to such errors is examined, and whenever necessary, new coding constraints are proposed to ensure unique reconstruction.**

I. Introduction

As we progress through this digital age, our rate of data generation continues to rise unhindered, and with it, so do our storage requirements. Since current data storage media are not particularly advantageous in regard to longevity or density, several molecular storage techniques [1]–[9] have been proposed. The work in [1] involving synthetic polymer-based storage systems appears to be especially favorable, given its promise of efficient synthesis, low read latency and cost. Under this paradigm, a string of information bits is encoded into a chain of molecules linked by means of phosphate bonds, such that the component molecules may only assume one of two significantly differing masses, which indicate the bits 0 and 1 respectively. The stored data can be read out by employing a tandem mass (MS/MS) spectrometer, which essentially splits the synthesized polymer at the phosphate linkages and outputs the masses of the resulting fragments. In this manner, the user is given access to the masses of all substrings in the encoded string.

A previous work [10] dealt with the problem of reconstructing a binary string from such an MS/MS readout, under the following modeling assumptions.

Assumption 1. Masses of the component molecules are chosen such that one can always uniquely infer the *composition*, i.e., the number of 0s and 1s forming a certain fragment, from its mass.

Assumption 2. While fragmenting a polymer for the purpose of mass spectrometry analysis, the masses of all of its constituent substrings are observed with identical frequency.

This setting simplifies the recovery of the original information string into the problem of binary string reconstruction from its composition multiset. More plainly, the reconstruction process now involves determining the binary string from a set of compositions of all of its substrings of each possible length. It is worth noting that a string and its reversal have identical sets of substring compositions, thus preventing us from distinguishing between them.

While the authors of [10] mainly focused on string lengths that ensured unique reconstruction from a composition multiset, subsequent works [11]–[13] extended this research by building a code that allows for unique reconstruction of each member codeword from its composition multiset alone, regardless of the string length. Similar coding constraints were also proposed to also correct possible errors in the composition multiset. The work in [14] takes a step further by dealing with the recovery of multiple strings from the mass spectrometry readout of a mixture of synthesized polymers.

Since the errors introduced during an MS/MS readout are normally context-dependent, we devote this work to the extension of the error models considered in [11], [12]. Specifically, we investigate the impact of inserting and deleting one or more compositions on the reconstructability of the encoded strings. Furthermore, new coding constraints are proposed to enable the correction of such errors. We also consider a special kind of substitution error, namely a *skewed substitution error*. Such errors are motivated by imperfect fragmentations of a given polymer during the MS/MS readout process, as a result of which the observed molecular mass of a shorter monomer chain is lower than what the true mass of its perfectly fragmented version would have been. In this scenario, errors occur only in one direction, i.e., the measured mass can only be lower than the true mass. An error-correcting scheme is

This work has been supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant Agreement No. 801434).

also suggested for this setting.

Due to space limitations, some proofs and examples are omitted and can be found in the long version [15].

II. Preliminaries

Let $\boldsymbol{s} = s_1 s_2 \dots s_n$ denote a binary string of n bits. Any substring $s_i \dots s_j$ where $i \leq j$, may be indicated by $\boldsymbol{s}_i^j$. The *composition* of this substring, denoted by $c(\boldsymbol{s}_i^j)$, is said to be $0^z 1^w$, where z and w refer to the number of 0s and 1s in $\boldsymbol{s}_i^j$ respectively, such that $z + w = j - i + 1$. We also define $C_k(\boldsymbol{s})$ as the set of compositions $\boldsymbol{s}_i^{i+k-1}$, for $1 \leq i \leq n-k+1$. Evidently, $C_k(\boldsymbol{s})$ should contain $n-k+1$ compositions.

Upon combining the multisets for all $1 \leq k \leq n$, we obtain the *composition multiset* of $\boldsymbol{s}$,

$$C(\boldsymbol{s}) = \bigcup_{k \in [n]} C_k(\boldsymbol{s}),$$

where $[n] = \{1, \dots, n\}$. The authors of [10] determined string lengths for which unique reconstruction (up to reversal) from such sets is possible. For the remaining string lengths, a bivariate generating polynomial representation was exploited to find strings that are equicomposable with a given string. Here, two distinct strings $\boldsymbol{s}, \boldsymbol{t} \in \{0,1\}^n$ are said to be *equicomposable* if a common composition multiset is shared, i.e., $C(\boldsymbol{s}) = C(\boldsymbol{t})$.

A code $\mathcal{C}$ is called a *composition-reconstructable code* if for all $\boldsymbol{s}, \boldsymbol{t} \in \mathcal{C}$, it holds that $C(\boldsymbol{s}) \neq C(\boldsymbol{t})$. For all n, we let $A(n)$ denote the size of the largest composition reconstructable code of length n. Since composition multisets are identical for a binary string and its reversal, it holds that

$$A(n) \leq 2^{\lceil \frac{n}{2} \rceil} + \frac{1}{2}(2^n - 2^{\lceil \frac{n}{2} \rceil}) = 2^{n-1} + 2^{\lceil \frac{n}{2} \rceil - 1},$$

where $2^{\lceil \frac{n}{2} \rceil}$ describes the number of palindromic strings of length n, and [10] determined string lengths n where it is possible to achieve this bound with equality. Specifically, it was shown that binary strings of length ≤ 7, of length one less than a prime, or of length one less than twice a prime, are uniquely reconstructable up to reversal. For all other values of n, it is not possible to achieve the aforementioned bound, and thus a code must be formulated. The composition-reconstructable code proposed in [11], [12] for even codeword lengths n is stated as follows:

$$\begin{aligned} \mathcal{S}_R(n) = \big\{ & \boldsymbol{s} \in \{0,1\}^n, s_1 = 0, s_n = 1, \text{ and} \\ & \exists I \subset \{2, \dots, n-1\} \text{ such that} \\ & \quad \text{for all } i \in I, s_i \neq s_{n+1-i}, \\ & \quad \text{for all } i \notin I, s_i = s_{n+1-i}, \\ & \boldsymbol{s}_{[n/2] \cap I} \text{ is a Catalan-Bertrand string.} \big\} \end{aligned} \tag{1}$$

In this context, a *Catalan-Bertrand string* refers to any binary vector wherein each prefix contains strictly more 0s than 1s.

The decoder of this code recovers a string from its composition multiset by employing the approach outlined in [10], [11]. We recapitulate a few underlying principles of this process since they help to formulate coding constructions for the newer error models involving insertions and deletions.

The algorithm begins by deducing the following sequence that characterizes the string to be recovered, say $\boldsymbol{s} \in \{0,1\}^n$,

$$\boldsymbol{\sigma}_s = (\sigma_1, \dots, \sigma_{\lceil n/2 \rceil}),$$

where $\sigma_i = \mathrm{wt}(s_i s_{n-i+1})$ for $i \in \{1, \dots, \lfloor n/2 \rfloor\}$. For odd n, we set $\sigma_{\lceil \frac{n}{2} \rceil} = \mathrm{wt}(s_{\lceil \frac{n}{2} \rceil})$, i.e., the weight of the central bit.

For any string $\boldsymbol{s}$, one can compute $\boldsymbol{\sigma}_s$ from its composition multiset by exploiting the concept of *cumulative weights*, which are defined for each multiset $C_k(\boldsymbol{s})$ as

$$w_k(\boldsymbol{s}) = \sum_{0^z 1^w \in C_k(\boldsymbol{s})} w.$$

It is easy to verify that for all $k \leq \lceil \frac{n}{2} \rceil$, these weights obey the following symmetry relation [11].

$$w_k(\boldsymbol{s}) = w_{n-k+1}(\boldsymbol{s}), \quad \forall\, k \in [n]. \tag{2}$$

In light of this, the multisets C_i and C_{n-i+1} are henceforth called *symmetric*. For notational convenience, we also define

$$\widetilde{C}_i(\boldsymbol{s}) = C_i(\boldsymbol{s}) \cup C_{n-i+1}(\boldsymbol{s}).$$

Furthermore, the cumulative weights of the multisets of a string $\boldsymbol{s}$, are related to the elements of $\boldsymbol{\sigma}_s$ as follows:

$$w_k(\boldsymbol{s}) = k w_1(\boldsymbol{s}) - \sum_{i=1}^{k-1} i \sigma_{k-i}. \tag{3}$$

Thus, by progressively using this equation for $k \in \{2, \dots, \lceil \frac{n}{2} \rceil\}$, we can determine $\boldsymbol{\sigma}_s$.

III. New Error Models

The subsequent sections explore error models that involve corrupting a valid composition multiset via the insertion of compositions or deletion of one or more multisets.

Definition 1. *An **asymmetric multiset deletion** is said to have occurred in the composition multiset $C(\boldsymbol{s})$ of a string $\boldsymbol{s} \in \{0,1\}^n$, if for some $i \in [n]$, the multiset $C_i(\boldsymbol{s})$ is entirely missing, while $C_{n-i+1}(\boldsymbol{s})$ is not corrupted.*

Definition 2. *A **pair of symmetric multiset deletions** is said to have occurred in the composition multiset $C(\boldsymbol{s})$ of a string $\boldsymbol{s} \in \{0,1\}^n$, if for some $i \in [n]$ such that $i \neq n-i+1$, the multisets $C_i(\boldsymbol{s})$ and $C_{n-i+1}(\boldsymbol{s})$ are entirely eliminated.*

Definition 3. *A composition multiset $C(\boldsymbol{s})$ of a string $\boldsymbol{s} \in \{0,1\}^n$ is said to have suffered a **composition insertion error**, if for some $i \in [n]$ the multiset $C_i(\boldsymbol{s})$ contains $n-i+2$ compositions, i.e., an unknown and invalid composition has been registered.*

This work primarily studies the aforementioned error models and proposes new coding constraints to permit the correction of such errors. Additionally in Section IV, we establish an equivalence between codes that correct composition insertions and composition deletions. Consequently, we restrict our attention to the latter for the remainder of this paper.

To this end, we first attempt to form a composition-reconstructable code that allows the correction of t asymmetric multiset deletions. Specifically, a code $\mathcal{S}_{DA}^{(t)}$ is termed as a *t-asymmetric multiset deletion composition code*, if for all $\boldsymbol{s}$, $\boldsymbol{v} \in \mathcal{S}_{DA}^{(t)}$, there exists no $\mathcal{I} \subseteq [n]$ with $|\mathcal{I}| \leq t$ such that for all $i \in \mathcal{I}$,

$$\begin{aligned} C_i(\boldsymbol{s}) &\neq C_i(\boldsymbol{v}), \\ C_{n-i+1}(\boldsymbol{s}) &= C_{n-i+1}(\boldsymbol{v}), \\ C_j(\boldsymbol{s}) &= C_j(\boldsymbol{v}) \quad \forall\, j \in [n] \setminus \mathcal{I}. \end{aligned} \tag{4}$$

Such a construction is discussed in Section V-A.

Subsequently in Section V-B, we investigate the case when a pair of symmetric multisets is deleted. In this context, a code $\mathcal{S}_{DS}^{(t)}$ is termed as a *t-symmetric multiset deletion composition code*, if for all $\boldsymbol{s}$, $\boldsymbol{v} \in \mathcal{S}_{DS}^{(t)}$, there exists no $\mathcal{I} \subseteq \left[\left\lceil \frac{n}{2} \right\rceil\right]$ with $|\mathcal{I}| \leq t$ such that

$$\begin{aligned} \widetilde{C}_i(\boldsymbol{s}) &\neq \widetilde{C}_i(\boldsymbol{v}), \quad \forall\, i \in \mathcal{I} \\ C_i(\boldsymbol{s}) &= C_i(\boldsymbol{v}), \quad \forall\, i \in \left[\left\lceil \frac{n}{2} \right\rceil\right] \setminus \mathcal{I}. \end{aligned}$$

We discover that the code $S_R(n)$ can correct the deletion of a single pair of symmetric multisets, and as a consequence, can also correct the substitution of a single composition in its complete composition multiset.

In addition to multiset deletions, we consider another error model in Section VI, that involves a special kind of substitution, defined as follows.

Definition 4. *A composition multiset $C(\boldsymbol{s})$ of the string $\boldsymbol{s} \in \{0,1\}^n$ is said to have suffered an* ***asymmetric skewed substitution error****, if for some $i \in [n]$, a single composition of the multiset $C_i(\boldsymbol{s})$ is replaced with one of a lower Hamming weight, such that the symmetric counterpart $C_{n-i+1}(\boldsymbol{s})$ remains unaffected.*

Formally, a code $\mathcal{C}'^{(t)}$ is referred to as a *t-asymmetric skewed composition code*, if for all $\boldsymbol{s}$, $\boldsymbol{v} \in \mathcal{C}'^{(t)}$, there exists no $\mathcal{I} \subseteq [n]$ with $|\mathcal{I}| \leq t$ such that for all $i \in \mathcal{I}$,

$$\begin{aligned} C_i(\boldsymbol{s}) &\neq C_i(\boldsymbol{v}), \\ C_{n-i+1}(\boldsymbol{s}) &= C_{n-i+1}(\boldsymbol{v}), \\ w_i(\boldsymbol{s}) &< w_{n-i+1}(\boldsymbol{s}), \\ C_j(\boldsymbol{s}) &= C_j(\boldsymbol{v}), \quad \forall\, j \in [n] \setminus \mathcal{I}. \end{aligned}$$

We also prove in Theorem 2 of Section VI that a code $\mathcal{S}_{DA}^{(t)}(n)$ that can correct at most t asymmetric multiset deletions, can also rectify up to t skewed asymmetric substitution errors in its composition multiset.

IV. Code Equivalence: Insertion and Deletion of Multisets

In this section, we demonstrate how codes which can correct the deletion of an arbitrary group of t multisets, can also correct the occurrence of insertion errors in those t multisets [1].

Lemma 1. *A code can correct the deletion of t composition multisets if and only if it can correct any number of composition insertion errors in those t multisets.*

Proof. We prove this by contradiction. Let there be two binary strings $\boldsymbol{s}, \boldsymbol{v} \in \{0,1\}^n$ such that:

$$\begin{aligned} C(\boldsymbol{s}) &\neq C(\boldsymbol{v}) \\ D_t(\boldsymbol{s}) \cap D_t(\boldsymbol{v}) &\neq \emptyset. \end{aligned} \tag{5}$$

where $D_t(\boldsymbol{s})$ constitutes all $\boldsymbol{u} \in \mathcal{S}_R(n)$ that $\boldsymbol{s}$ becomes equicomposable with, upon the deletion of at most t multisets, i.e.,

$$\begin{aligned} D_t(\boldsymbol{s}) = \{\boldsymbol{u} \in \mathcal{S}_R(n) : \exists\, \mathcal{I} \subseteq [n],\ |\mathcal{I}| \leq t, \\ \bigcup_{i \in [n] \setminus \mathcal{I}} C_i(\boldsymbol{s}) = \bigcup_{i \in [n] \setminus \mathcal{I}} C_i(\boldsymbol{u})\}. \end{aligned}$$

Equation (5) implies that at least $n-t$ composition multisets of $\boldsymbol{s}$ and $\boldsymbol{v}$ are identical. In other words, $\boldsymbol{s}$ and $\boldsymbol{v}$ have at most t differing multisets C_i for $i \in \mathcal{I} \subset [n]$, and when these specific multisets disappear from both of their composition multiset, $\boldsymbol{s}$ and $\boldsymbol{v}$ become indistinguishable. Let these differing multisets correspond to substring lengths $i_1, i_2, \ldots, i_t$. We may write:

$$\bigcup_{j \in [n] \setminus \{i_1, \ldots, i_t\}} C_j(\boldsymbol{s}) = \bigcup_{j \in [n] \setminus \{i_1, \ldots, i_t\}} C_j(\boldsymbol{v}).$$

Upon performing a set union operation on both sides of the previous equation with $\bigcup_{i \in \{i_1, \ldots, i_t\}} C_i(\boldsymbol{s}) \cup C_i(\boldsymbol{v})$, we get

$$\begin{aligned} &\bigcup_{i \in \{i_1, \ldots, i_t\}} (C_j(\boldsymbol{v}) \backslash C_j(\boldsymbol{s})) \cup \bigcup_{j \in [n]} C_j(\boldsymbol{s}) \\ = &\bigcup_{i \in \{i_1, \ldots, i_t\}} (C_j(\boldsymbol{s}) \backslash C_j(\boldsymbol{v})) \cup \bigcup_{j \in [n]} C_j(\boldsymbol{v}). \end{aligned}$$

This effectively means that if the multisets $C_{i_1}(\boldsymbol{s}), \ldots, C_{i_t}(\boldsymbol{s})$ are corrupted by the insertion of some specific erroneous compositions, then this corrupted composition multiset may correspond to both $\boldsymbol{s}$ and $\boldsymbol{v}$, and vice-versa. Thus,

$$I_t(\boldsymbol{s}) \cap I_t(\boldsymbol{v}) \neq \emptyset, \tag{6}$$

where $I_t(\boldsymbol{s})$ denotes the set of all codewords $\boldsymbol{u} \in \mathcal{S}_R(n)$ whose composition multisets, upon suffering any number of insertion errors in at most t distinct multisets, resemble $C(\boldsymbol{s})$ after corruption by certain composition insertions in those affected multisets. In other words, at least $n-t$ distinct multisets of $\boldsymbol{s}$ and $\boldsymbol{u}$ are identical. As a consequence,

$$\begin{aligned} I_t(\boldsymbol{s}) =& D_t(\boldsymbol{s}) \\ =& \{\boldsymbol{u} \in \mathcal{S}_R(n) : \exists\, \mathcal{I} \subseteq [n],\ |\mathcal{I}| \leq t, \\ & \forall\, i \in [n] \setminus \mathcal{I},\ C_i(\boldsymbol{s}) = C_i(\boldsymbol{u})\} \end{aligned}$$

□

Owing to this result, we deem it sufficient to focus on error models involving the deletion of one or more multisets. The subsequent sections examine how multiset deletions affect the reconstructability of an encoded string drawn from $\mathcal{S}_R(n)$. Similar to [11], we categorize such deletion errors into two major settings, namely symmetric and asymmetric.

[1] Error models involving both insertions and deletions do not fall under this equivalence.

V. Multiset Deletion Composition Codes

This section discusses codes that are capable of correcting the deletion of one or more multisets under the asymmetric and symmetric settings [see Definition 1 and 2].

A. Asymmetric setting

We begin by considering an error model where a complete multiset $C_k(\boldsymbol{s})$ can be deleted from the composition multiset $C(\boldsymbol{s})$. This is formally referred to as a single asymmetric multiset deletion. We investigate whether $\mathcal{S}_R(n)$ guarantees unique recoverability under this model.

Lemma 2. *[15] The code $\mathcal{S}_R(n)$ is a single asymmetric multiset deletion composition code.*

It is shown in [15] that when multiple asymmetric multiset deletions occur, $\mathcal{S}_R(n)$ can no longer correct them. To remedy this, we generalize $\mathcal{S}_R(n)$ to a new construction $\mathcal{S}_{DA}^{(t)}(n)$, which is a t-asymmetric multiset deletion composition code. This code is inspired from the following construction in [11], which is an adaption of the code in (1) by increasing the prefix and suffix lengths.

$$\begin{aligned}\mathcal{S}_R^{(t)}(n) = \big\{ \boldsymbol{s} \in \{0,1\}^n, \boldsymbol{s}_1^t = \mathbf{0}, \boldsymbol{s}_{n-t+1}^n = \mathbf{1}, &\text{ and} \\ \exists I \subset \{t+1, \dots, n-t\} &\text{ such that} \\ \text{for all } i \in I, s_i \neq s_{n+1-i}&, \\ \text{for all } i \notin I, s_i = s_{n+1-i}&, \\ \boldsymbol{s}_{[n/2] \cap I} \text{ is a Catalan-Bertrand string}&\big\}.\end{aligned} \tag{7}$$

This code can correct at most t substitution errors under an error model that allows up to one substitution error in the multiset $\widetilde{C}_i$ for $i \in [n]$. We now propose the following t-asymmetric multiset deletion composition code.

Construction 1:

$$\begin{aligned}\mathcal{S}_{DA}^{(t)}(n) = \big\{ \boldsymbol{s} \in \{0,1\}^n : s_1 = 0, s_n = 1, &\text{ and} \\ \exists I \subset \{2, \dots, \frac{n}{2}\},\ |I| \geq t, &\text{ such that} \\ \forall\, i \in I, s_i \neq s_{n+1-i}&, \\ \text{and } \forall i \notin I, s_i = s_{n+1-i}&, \\ \boldsymbol{s}_{[n/2] \cap I} \text{ is a string wherein each}& \\ \text{prefix has at least } t \text{ more 0s than 1s}&\big\}.\end{aligned} \tag{8}$$

Evidently, this construction is similar to $\mathcal{S}_R^{(t)}(n)$ in that it requires at least t 0s in $\boldsymbol{s}_1^{n/2}$ and at least t 1s in $\boldsymbol{s}_{n/2+1}^n$, however their locations are not necessarily restricted, unlike $\mathcal{S}_R^{(t)}(n)$. A similar construction for odd n also exists.

Theorem 1. *[15] The code $\mathcal{S}_{DA}^{(t)}(n)$ is a t-asymmetric multiset deletion composition code.*

Proof sketch. Consider any $\boldsymbol{s} \in \mathcal{S}_{DA}^{(t)}(n)$. We begin with the observation that despite the deletion of t asymmetric multisets from $C(\boldsymbol{s})$, $\boldsymbol{\sigma}_s$ can be recovered, since according to (2), the cumulative weight of each deleted multiset is equal to that of its symmetric counterpart, which is preserved. Hence, if there exist two strings $\boldsymbol{s}$, $\boldsymbol{v} \in \mathcal{S}_{DA}^{(t)}(n)$ that are "confusable" after the deletion of t asymmetric multisets, i.e., for some $\mathcal{I} \subseteq [n]$ with $|\mathcal{I}| \leq t$, $\boldsymbol{s}$ and $\boldsymbol{v}$ satisfy (4) for all $i \in \mathcal{I}$, then it must hold that $\boldsymbol{\sigma}_s = \boldsymbol{\sigma}_v$.

Next, we prove that if all of the t deleted multisets are consecutive, then any $\boldsymbol{s}, \boldsymbol{v} \in \mathcal{S}_{DA}^{(t)}(n)$ that satisfy $\boldsymbol{\sigma}_s = \boldsymbol{\sigma}_v$, differ by at least two compositions in a minimum of $t+1$ multisets [15, Lemma 5]. The approach taken to accomplish this is similar to that used for [11, Lemma 4], i.e., we create a set $\mathcal{V}_s$ containing all strings $\boldsymbol{v}$ that fulfill two particular conditions. Firstly, it must hold that $\boldsymbol{\sigma}_s = \boldsymbol{\sigma}_v$, and secondly, if $(\boldsymbol{s}_1^k, \boldsymbol{s}_{n-k+1}^n) = (\boldsymbol{v}_1^k, \boldsymbol{v}_{n-k+1}^n)$, then for all $i \in \{1, \dots, t+1\}$,

$$|C_{n-k-i}(\boldsymbol{s}) \setminus C_{n-k-i}(\boldsymbol{v})| \leq 2.$$

We are interested specifically in $\mathcal{V}_s$, since it accounts for all such $\boldsymbol{v}$ that may be confused with $\boldsymbol{s}$ during the reconstruction algorithm, as mentioned in Section II. After building $\mathcal{V}_s$, we are able to infer that there exists no such $\boldsymbol{v} \in \mathcal{S}_{DA}^{(t)}(n) \cap \mathcal{V}_s$ that achieves the conditions $(\boldsymbol{s}_1^k, \boldsymbol{s}_{n-k+1}^n) = (\boldsymbol{v}_1^k, \boldsymbol{v}_{n-k+1}^n)$, $s_{k+1} \neq v_{k+1}$ and $C_{n-k-t-1}(\boldsymbol{s}) = C_{n-k-t-1}(\boldsymbol{v})$ simultaneously. In other words, if the multisets $C_{n-k-1}(\boldsymbol{s}), \dots, C_{n-k-t}(\boldsymbol{s})$ are deleted from $C(\boldsymbol{s})$, we can still recover $\boldsymbol{s}$ uniquely, since there exists no such $\boldsymbol{v} \in \mathcal{S}_{DA}^{(t)}(n)$ that satisfies (4) for all $i \in \mathcal{I}$, where $\mathcal{I} \subseteq [n]$ with $|\mathcal{I}| \leq t$.

For the case when the not all of deleted multisets are consecutive, the proof follows similarly. □

We also bound the required redundancy of $\mathcal{S}_{DA}^{(t)}(n)$ as follows.

Lemma 3. *The code $\mathcal{S}_{DA}^{(t)}(n)$ requires at most $\frac{1}{2}\log(n-2t) + 2t + 3$ bits of redundancy.*

Proof. We refer to (8) and also recount from [11] that $\frac{1}{2}\binom{2h}{h}$ indicates the number of all strings of length $2h$ wherein every prefix of which contains strictly more 0s than 1s. For odd lengths $2h+1$, this term serves as a lower bound. Similarly, to count all strings $\boldsymbol{s} \in \{0,1\}^p$ wherein each prefix (of length exceeding t) contains at least t more 0s than 1s, we simply note that such strings satisfy $\boldsymbol{s}_1^{t-1} = \mathbf{0}$ and $\boldsymbol{s}_t^p$ should be a standard Catalan-Bertrand string. By virtue of this, we derive a lower bound on the dimension of the codebook.

$$|\mathcal{S}_{DA}^{(t)}(n)| \geq \sum_{i=t}^{n/2-1} 2^{n/2-2-i} \binom{n/2-1}{i} \binom{i-t+1}{\lfloor (i-t+1)/2 \rfloor}.$$

After some algebraic manipulations, we deduce that the number of redundant bits necessary is at most $\frac{1}{2}\log(n-2t) + 2t + 3$. □

B. Symmetric setting

As mentioned in Section III, errors under this category occur in such a way that the affected multisets occur in pairs. Specifically, we focus on the case when a single pair of mutually symmetric multisets is inaccessible[2].

[2] Constructions correcting the deletions of multiple pairs of symmetric multisets will be presented later.

Lemma 4. *[15] The code $\mathcal{S}_R(n)$ is a single symmetric multiset deletion composition code.*

The previous result reveals that the codebook $\mathcal{S}_R(n)$ is sufficiently robust to correct the deletion of a single pair of symmetric multisets. Consequently, if a single composition is substituted in $C(\boldsymbol{s})$ where $\boldsymbol{s} \in \mathcal{S}_R(n)$, then there occurs a mismatch between the cumulative weights of the specific multiset affected, say $C_i(\boldsymbol{s})$, and its symmetric counterpart $C_{n-i+1}(\boldsymbol{s})$. It is thus possible to identify $\widetilde{C}_i(\boldsymbol{s})$. Now if we delete $\widetilde{C}_i(\boldsymbol{s})$ entirely, Lemma 4 tells us that $\boldsymbol{s}$ is still uniquely recoverable. Thus, $\mathcal{S}_R(n)$ can correct a single composition error just like $S^{(1)}_{CA}(n)$ [11], [12].

VI. Asymmetric Skewed Composition codes

In this section, we confine our focus to the correction of asymmetric skewed substitution errors [see Definition 4].

Lemma 5. *Consider any $\boldsymbol{s} \in \mathcal{S}_R(n)$. Given that there occurs a single skewed substitution error in its composition multiset, one can uniquely recover $\boldsymbol{s}$.*

Proof. In the following, we let the corrupted composition multiset be denoted by $C'(\boldsymbol{s}) = \bigcup_{i\in[n]} C'_i(\boldsymbol{s})$.

Case 1. n is even.

Given $C'(\boldsymbol{s})$, it is easy to identify the corrupted composition multiset $C'_k(\boldsymbol{s})$, since the following relation only holds for k:

$$w'_k < w'_{n-k+1}. \tag{9}$$

We recall from (2) that these quantities should ideally be equal. If we now delete all elements of $C'_k(\boldsymbol{s})$ from $C'(\boldsymbol{s})$, Lemma 2 tells us that $\boldsymbol{s}$ is still uniquely recoverable.

Case 2. n is odd.

Using the arguments of the preceding case, we reach the same conclusion for an odd n, when the affected multiset is $C'_k(\boldsymbol{s})$ where $\lceil n/2\rceil < k \leq n$, because in these cases, an uncorrupted distinct symmetric counterpart $C'_{n-k+1}(\boldsymbol{s})$ that gives us the true cumulative weight, exists. Thus $\boldsymbol{\sigma}_s$ can be accurately recovered.

If $k = \lceil n/2\rceil$, this is no longer true since the multiset $C_{\lceil n/2\rceil}(\boldsymbol{s})$ is its own symmetric counterpart. We note from (3) that this normally helps us determine $\sigma_{\lceil \frac{n}{2}\rceil -1}$ and in turn, the bits $(s_{\lceil n/2\rceil-1}, s_{\lceil n/2\rceil+1})$. We also observe from [15, Lemma 2] that when these bits are assigned incorrectly, inconsistencies with the multiset $C_{\lceil n/2\rceil-1}$ would arise, which are not permitted under the considered error model. Hence, we conclude that $\boldsymbol{s}$ can be recovered uniquely.

□

We now consider a more general error model involving multiple asymmetric skewed substitution errors, wherein each multiset pair $\widetilde{C}_i$, for any $i \in [n]$, may contain at most one skewed substitution and the total number of errors does not exceed t. In the following, we prove that the asymmetric t-multiset deletion-correcting code $\mathcal{S}^{(t)}_{DA}(n)$ is also robust to t asymmetric skewed substitutions.

Theorem 2. *Consider any $\boldsymbol{s} \in \mathcal{S}^{(t)}_{DA}(n)$. Given that there occur t skewed asymmetric substitution errors in its composition multiset, such that for all $1 \leq i \leq n$, $\widetilde{C}_i(\boldsymbol{s})$ contains at most one skewed substitution error, then one can uniquely recover $\boldsymbol{s}$.*

Proof. Since the error model allows at most one skewed substitution in a pair of symmetric multisets, all cumulative weights can be determined accurately. This is due to the fact that if a multiset $C_k(\boldsymbol{s})$ has been corrupted, we may write

$$w_k < w_{n-k+1}. \tag{10}$$

Hence, the cumulative weights can be correctly re-assigned and in turn $\boldsymbol{\sigma}_s$ can be recovered. The preceding inequality also allows to identify the affected multisets, the deletion of which would transform our problem of correcting t asymmetric skewed substitutions into reconstruction under the absence of t multisets. According to Theorem 1, unique reconstruction of $\boldsymbol{s}$ is perfectly possible, thus concluding our proof. □

As a consequence, $\mathcal{S}^{(t)}_{DA}(n)$ is also a t-asymmetric skewed composition code.

VII. Outlook

Several problems pertaining to string reconstruction under this data storage paradigm still remain open:

- The error model involving skewed substitutions under a symmetric setting is yet to be investigated. It would be interesting to know if there exists a suitable codebook offering a lower redundancy than that designed to correct standard substitution errors under the symmetric setting, as stated in [11].
- The problem of reconstructing strings from composition multisets, error-free or otherwise, could be extended to larger alphabets.
- Though some bounds on the maximum number of mutually equicomposable strings were stated in [10], bounds on the error ball sizes under the error models involving substitutions, insertions or deletions are still unknown. These could allow us to infer if the suggested code constructions are indeed optimal. For instance, our simulations reveal that a composition-reconstructable code requiring a lower redundancy than $\mathcal{S}_R(n)$ should exist. This makes intuitive sense since $\mathcal{S}_R(n)$ is designed for decoding efficiency, in that no backtracking is required during the reconstruction process of its strings [11], [12].
- One could also extend this research to the construct wherein bits are arranged in a circular fashion, on a ring.
- As pointed out in [10], a polynomial-time algorithm for the string reconstruction problem is yet to be found.

References

[1] A. Al Ouahabi, J.-A. Amalian, L. Charles, and J.-F. Lutz, "Mass spectrometry sequencing of long digital polymers facilitated by programmed inter-byte fragmentation," *Nature communications*, vol. 8, no. 1, p. 967, 2017.

[2] 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, p. 77, 2013.

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

[4] R. Heckel, G. Mikutis and R. N. Grass, "A characterization of the DNA data storage channel," *Scientific Reports*, vol. 9, no. 1, pp. 9663, 2019.

[5] C.N. Takahashi, B.H. Nguyen, K. Strauss and L. Ceze, "Demonstration of end-to-end automation of DNA data storage," *Scientific Reports*, vol. 9, no. 1, pp. 4998, 2019.

[6] S. H. T. Yazdi, Y. Yuan, J. Ma, H. Zhao, and O. Milenkovic, "A rewritable, random-access DNA-based storage system," *Scientific reports*, vol. 5, p. 14138, 2015.

[7] S. H. T. Yazdi, R. Gabrys, and O. Milenkovic, "Portable and error-free DNA-based data storage," *Scientific reports*, vol. 7, no. 1, p. 5011, 2017.

[8] S. K. Tabatabaei, B. Wang, N. B. M. Athreya, B. Enghiad, A. G. Hernandez, J.-P. Leburton, D. Soloveichik, H. Zhao, and O. Milenkovic, "DNA punch cards: Encoding data on native DNA sequences via topological modifications," *bioRxiv*, p. 672394, 2019.

[9] S. Tabatabaei, B. Wang, N. Athreya, B. Enghiad, A. Hernandez, C. Fields, J.-P. Leburton, D. Soloveichik, H. Zhao, and O. Milenkovic, "DNA punch cards for storing data on native DNA sequences via enzymatic nicking," *Nature Communications*, vol. 11, 12 2020.

[10] J. Acharya, H. Das, O. Milenkovic, A. Orlitsky, and S. Pan, "String reconstruction from substring compositions," *SIAM Journal on Discrete Mathematics*, vol. 29, no. 3, pp. 1340–1371, 2015.

[11] S. Pattabiraman, R. Gabrys and O. Milenkovic, "Coding for polymer-based data storage", *arXiv:2003.02121*, 2020

[12] S. Pattabiraman, R. Gabrys, and O. Milenkovic, "Reconstruction and error-correction codes for polymer-based data storage," *Proc. IEEE Information Theory Workshop*, Visby, Sweden, pp. 1–5, Aug., 2019.

[13] R. Gabrys, S. Pattabiraman, and O. Milenkovic, "Mass error-correction codes for polymer-based data storage," *Proc. IEEE International Symposium on Information Theory*, Los Angeles, CA, USA, pp. 25–30, Jun., 2020.

[14] R. Gabrys, S. Pattabiraman, and O. Milenkovic, "Reconstructing mixtures of coded strings from prefix and suffix compositions," *Proc. IEEE Information Theory Workshop*, Kanazawa, Japan, pp. 1–5, Oct. 2021.

[15] A. Banerjee, A. Wachter-Zeh, and E. Yaakobi, "Insertion and deletion correction in polymer-based data storage" (extended version of this paper), arXiv preprint arXiv:2201.08612, 2022.