# On Levenshtein's Reconstruction Problem Under Insertions, Deletions, and Substitutions

Maria Abu-Sini, *Student Member, IEEE*, and Eitan Yaakobi, *Senior Member, IEEE*

***Abstract*— The *sequence reconstruction problem* corresponds to the model in which a sequence from some code is transmitted over several noisy channels that produce distinct outputs. Then, the channels' outputs, received by the decoder, are used to recover the transmitted sequence, and the main problem under this paradigm is to calculate the minimum number of channels that enables unique reconstruction of the transmitted word. This problem is equivalent to finding the size of the largest intersection of channels' outputs sets received after transmitting distinct codewords. Motivated by the error behavior observed in DNA storage systems, the present work extends the study of the reconstruction model to the case in which a binary word is transmitted over channels prone to substitutions, insertions, and deletions. Furthermore, we also study the size of the error balls generated by either one deletion and at most a fixed number of substitutions or one insertion and at most one substitution in a binary word. For the case of only substitutions, we present a decoder of optimal complexity, which improves upon a recent construction of such a decoder. Lastly, a simplification of that decoder is studied in case there are more channels than the minimum required number.**

***Index Terms*— Sequence reconstruction, Levenshtein distance, synchronization channel.**

## I. Introduction

THE reconstruction problem studied by Levenshtein in [15] and [16] corresponds to a model in which a word is transmitted over several identical noisy channels that produce distinct outputs. Under the assumption that all of the channels are prone to the same errors pattern, all of the channels' outputs belong to a certain error ball surrounding the transmitted word. For a transmitted word $\boldsymbol{w}$, denote this error ball by $B(\boldsymbol{w})$. Moreover, assume there are $m$ channels. Since these channels produce distinct words, then their outputs form a size-$m$ subset of $B(\boldsymbol{w})$ that, as depicted in Fig. 1, may be used entirely to reconstruct the transmitted word $\boldsymbol{w}$. Hence, when dealing with such a model, one may ask whether it is possible to unambiguously reconstruct the transmitted word, in particular efficiently.

Levenshtein studied the reconstruction model in [15] and [16] for several errors patterns including substitutions, insertions, and deletions. Further advances were also achieved in [21] for insertions and in [5] for deletions. Moreover, problems related to the reconstruction model were addressed in [9] and [22]. More specifically, the connection between the reconstruction problem and associative memories was studied in [22]. In addition, the intersection of error balls, which is highly connected to the reconstruction model, was analyzed in [9] in order to asymptotically improve the Gilbert-Varshamov bound. More results for other general error graphs and metrics were obtained in [8], [11]–[13], [17], [18], [23].

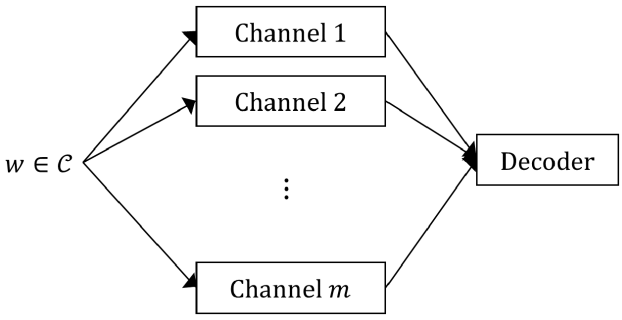


Fig. 1. The reconstruction model.

Recently, Levenshtein's reconstruction model has gained considerable significance due to the emergence of DNA storage solution. This storage solution was first implied to in [4] and suggests storing archival data in synthesized DNA strands. As we know, a DNA strand consists of a linear sequence of four types of nucleotides: Adenine (A), Cytonine (C), Guanine (G), and Thymine (T). Hence, in DNA storage systems, the process of storing the data begins with synthesizing DNA strands of the sequence that we want to store. However, due to biotechnological limitations, it is not possible to generate a single DNA strand of a given sequence [14]. Instead, many DNA strands with erroneous sequences are generated and represent the sequence we aimed to store. Thus, in such systems, a specific sequence is stored using many erroneous copies kept together in a DNA pool. Later, in order to retrieve the information, the sequences of the stored DNA strands are recovered by a process called *sequencing*. However, this process is also prone to errors and is usually preceded by PCR (polymerase chain reaction) amplification, which generates many copies of each DNA strand. In summary, in DNA storage systems, the original sequence is retrieved using many representing erroneous copies for two main reasons. First, because thousands of copies were synthesized, and second,

Manuscript received April 17, 2020; revised August 19, 2021; accepted September 1, 2021. Date of publication September 10, 2021; date of current version October 20, 2021. This work was supported in part by BSF under Grant 2018048. An earlier version of this paper was presented at the 2019 IEEE International Symposium on Information Theory [DOI: 10.1109/ISIT.2019.8849740]. *(Corresponding author: Eitan Yaakobi.)*

The authors are with the Department of Computer Science, Technion—Israel Institute of Technology, Haifa 3200003, Israel (e-mail: maria.as@cs.technion.ac.il; yaakobi@cs.technion.ac.il).

Communicated by L. Dolecek, Associate Editor for Coding Techniques.
Digital Object Identifier 10.1109/TIT.2021.3110710

because of the copies duplicated using PCR amplification [3]. Therefore, by considering the steps of storing and reading the data in DNA storage systems, one can recognize the reconstruction model described earlier. That is, investigating the reconstruction model is of sufficient interest for its potential application in DNA storage systems.

The rest of the paper is organized as follows. First, Section II presents the formal definition of the reconstruction problem. Then, motivated by the error behavior in DNA-based storage systems [6], which includes combinations of insertions, deletions, and substitutions of DNA strands' nucleotides, we extend the reconstruction problem for this family of error types. In Section III, we calculate the size of the single-deletion $t$-substitution ball, as well as the single-insertion single-substitution ball in Section IV. In Section V, we take the first steps in exploring the reconstruction problem for the combination of different error types and study the case of single-insertion single-substitution. Next, in Section VI, an optimal reconstructing algorithm for the case of only substitutions is presented. Lastly, in Section VII, it is studied how the decoding algorithms can be simplified in case there are more copies than the minimum number found by Levenshtein. Section VIII concludes the paper.

## II. Definitions and Preliminaries

In this section we review the reconstruction problem which was first proposed by Levenshtein in [15] and [16]. Let $V$ be the space of all possible words. In addition, assume that there are several identical noisy channels over which a word $\boldsymbol{w} \in V$ is transmitted, and that these channels produce distinct outputs. In this case, the set of all possible channel outputs forms an error ball surrounding the transmitted word $\boldsymbol{w}$, and this ball is denoted by $B(\boldsymbol{w})$. In this study it is assumed that the errors can be of several types and do not necessarily correspond to a specific distance metric . Thus, we keep the definition of the error ball as general as possible. For example, motivated by DNA-based storage systems, it will be assumed that each channel is prone to a combination of substitutions, deletions, and insertions. Then, the goal is to find the following value and algorithm.

- The minimum number of distinct outputs needed so that the transmitted word can be determined uniquely .
- An efficient algorithm that receives the channels' outputs and recovers the transmitted word.

More specifically, the following problems arise when studying the reconstruction model.

*Problem 1:* Given a code $\mathcal{C} \subseteq \{0,1\}^n$ and an error ball $B(\boldsymbol{w})$ surrounding a binary word $\boldsymbol{w}$. Find the largest intersection of two error balls surrounding distinct codewords, i.e. find the value of

$$\max_{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \mathcal{C}, \boldsymbol{w}_1 \neq \boldsymbol{w}_2} |B(\boldsymbol{w}_1) \cap B(\boldsymbol{w}_2)|. \tag{1}$$

Problem 1 was first initiated and studied by Levenshtein in [15] and [16] and is referred to here as the *reconstruction problem*. Assume the transmitted word $\boldsymbol{w}$ belongs to some code $\mathcal{C} \subseteq \{0,1\}^n$, then Levenshtein proved that the number of channels required for the existence of a successful unique decoder is strictly larger than the value in (1).

In Problem 1, the possibility of unique decoding is investigated. However, solving this problem does not necessarily reveal how to recover the transmitted word. In other words, given that the number of channels $m$ is greater than

$$\max_{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \mathcal{C}, \boldsymbol{w}_1 \neq \boldsymbol{w}_2} |B(\boldsymbol{w}_1) \cap B(\boldsymbol{w}_2)|,$$

it is guaranteed that unambiguous reconstruction of the transmitted word is possible, yet it is still not clear how to accomplish this task. Therefore, Problem 2 arises and asks to design a decoder that retrieves the transmitted word.

*Problem 2:* Let $\mathcal{C} \subseteq \{0,1\}^n$ be a code, $B(\boldsymbol{w})$ error ball surrounding a binary word $\boldsymbol{w}$ and

$$m \geqslant \max_{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \mathcal{C}, \boldsymbol{w}_1 \neq \boldsymbol{w}_2} |B(\boldsymbol{w}_1) \cap B(\boldsymbol{w}_2)| + 1.$$

Design an algorithm that gets as an input at least $m$ distinct words $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_m \in B(\boldsymbol{w})$ for a word $\boldsymbol{w} \in \mathcal{C}$, and returns $\boldsymbol{w}$.

Since a decoder from Problem 2 gets at least

$$\max_{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \mathcal{C}, \boldsymbol{w}_1 \neq \boldsymbol{w}_2} |B(\boldsymbol{w}_1) \cap B(\boldsymbol{w}_2)| + 1$$

channels' outputs, then under the assumption that all of the channels' outputs are of the same length $k$, the decoder reads in the worst case at least

$$k \cdot \left( \max_{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \mathcal{C}, \boldsymbol{w}_1 \neq \boldsymbol{w}_2} |B(\boldsymbol{w}_1) \cap B(\boldsymbol{w}_2)| + 1 \right)$$

bits, i.e., its run-time complexity is

$$\Omega\left( k \cdot \left( \max_{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \mathcal{C}, \boldsymbol{w}_1 \neq \boldsymbol{w}_2} |B(\boldsymbol{w}_1) \cap B(\boldsymbol{w}_2)| + 1 \right) \right).$$

Therefore, a further goal that one may aim to achieve is presented in Problem 3 and focuses on efficient algorithms. That is, Problem 3 asks to design reconstructing algorithms that receive the minimum number of channels' outputs and have minimum oder of run-time complexity.

*Problem 3:* Let $\mathcal{C} \subseteq \{0,1\}^n$ be a code, $B(\boldsymbol{w})$ error ball surrounding a binary word $\boldsymbol{w}$ such that all of the words in $\bigcup_{\boldsymbol{c} \in \mathcal{C}} B(\boldsymbol{c})$ are of the same length $k$, and

$$m = \max_{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \mathcal{C}, \boldsymbol{w}_1 \neq \boldsymbol{w}_2} |B(\boldsymbol{w}_1) \cap B(\boldsymbol{w}_2)| + 1.$$

Determine whether there exists an algorithm of run-time complexity

$$\Theta(m \cdot k)$$

that gets as an input $m$ distinct words $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_m \in B(\boldsymbol{w})$ for a word $\boldsymbol{w} \in \mathcal{C}$, and returns $\boldsymbol{w}$. If there exists such a decoder, then design one.

Problems 1, 2, and 3 may be tackled for several error models. For example, one can solve these problems under the assumption that $B(\boldsymbol{w})$ is an error ball resulting from at most $t_1$ substitutions, $t_2$ deletions, and $t_3$ insertions, where $t_1, t_2, t_3 \geqslant 0$.

Furthermore, investigating these problems is indeed of sufficient interest as they are motivated by the DNA-based storage systems. Recall that, in such systems, recovering the stored sequence is done by sequencing many erroneous strands. Thus, we may consider the noisy strands as channels in

which a sequence is transmitted. Hence, the value calculated in Problem 1 determines the minimum number of distinct noisy strands needed to guarantee successful unique decoding. Moreover, solutions to Problems 2 and 3 provide algorithms for retrieving the stored sequence using the erroneous strands.

In [5], [15], [16], and [21], Problems 1, 2, and 3 were tackled under the assumption that all of the channels are prone to a specific type of errors, i.e., either substitutions, insertions , or deletions. However, according to [6], combinations of errors might occur in one DNA strand. Therefore, studying Problems 1, 2, and 3 for combinations of errors is of high significance. The present paper expands existing results regarding the reconstruction model in the binary case, especially when the channels are prone to combinations of errors. However, among the most preliminary obstacles that should be tackled in order to achieve this goal we have the following problem, which asks for the size of the error balls resulting from combinations of errors.

*Problem 4:* Given $\boldsymbol{w} \in \{0,1\}^*$ and error ball $B(\boldsymbol{w})$, find $|B(\boldsymbol{w})|$.

In the rest of this section we present the notation used in the paper along with existing solutions to Problems 1, 2, and 3. These solutions were achieved under the assumption that the channels are prone to either substitutions, insertions , or deletions. Since our contribution is restricted to the binary case, we choose to present existing results for the binary case solely, even though some of them were achieved for every alphabet size.

In this paper, for $n, m \in \mathbb{N}$ such that $n \leqslant m$, $[n, m]$ stands for the set $\{n, n+1, \ldots, m\}$. Moreover, given a word $\boldsymbol{w} \in \{0,1\}^*$, the number of runs in $\boldsymbol{w}$ is denoted by $r(\boldsymbol{w})$. Next, let $\boldsymbol{w}_1, \boldsymbol{w}_2$ be two binary words of the same length. Then, the Hamming distance between $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ is denoted by $d_H(\boldsymbol{w}_1, \boldsymbol{w}_2)$. Similarly, the Levenshtein distance between $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$, which is one half of the minimum number of insertions and deletions required to convert $\boldsymbol{w}_1$ to $\boldsymbol{w}_2$, is denoted by $d_L(\boldsymbol{w}_1, \boldsymbol{w}_2)$.

Given a binary word $\boldsymbol{w}$, the notation $B_{t_1,t_2,t_3}(\boldsymbol{w})$ stands for the set of words obtained by at most $t_1$ substitutions and exactly $t_2$, $t_3$ deletions, insertions in $\boldsymbol{w}$, respectively. Moreover, let $B^*_{t_1,t_2,t_3}(\boldsymbol{w})$ be the set of words obtained by at most $t_1$, $t_2$, $t_3$, substitutions, deletions, insertions in $\boldsymbol{w}$, respectively. That is, in $B_{t_1,t_2,t_3}(\boldsymbol{w})$ the number of insertions, deletions is restricted to be exactly $t_2, t_3$, respectively. However, in $B^*_{t_1,t_2,t_3}(\boldsymbol{w})$ at most $t_2, t_3$ deletions, insertions occur, respectively. Given these definitions, we emphasize that the present work addresses combinations of substitutions with either insertions or deletions only, leaving combinations of insertions and deletions for future work. Therefore, studying a combination of exactly $t_2$ deletions with at most $t_1$ substitutions is sufficient to receive results regarding the case of at most $t_2$ deletions and at most $t_1$ substitutions. Mathematically speaking, given a binary word $\boldsymbol{w}$, it holds that

$$B^*_{t_1,t_2,0}(\boldsymbol{w}) = \bigcup_{i=0}^{t_2} B_{t_1,i,0}(\boldsymbol{w}).$$

Therefore, finding the size of $B_{t_1,i,0}(\boldsymbol{w})$ for every $i \in \mathbb{N}$ determines the size of $B^*_{t_1,t_2,0}(\boldsymbol{w})$ for every $t_2$. Similarly, calculating the value of

$$\max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \{0,1\}^n \\ \boldsymbol{w}_1 \neq \boldsymbol{w}_2}} \left| B_{t_1,t_2,0}(\boldsymbol{w}_1) \cap B_{t_1,t_2,0}(\boldsymbol{w}_2) \right|$$

is sufficient in order to find the value of

$$\max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \{0,1\}^n \\ \boldsymbol{w}_1 \neq \boldsymbol{w}_2}} \left| B^*_{t_1,t_2,0}(\boldsymbol{w}_1) \cap B^*_{t_1,t_2,0}(\boldsymbol{w}_2) \right|.$$

In addition, for the same analysis, results regarding the case of exactly $t_3$ insertions and at most $t_1$ substitutions can be easily extended to the case of at most $t_3$ insertions and at most $t_1$ substitutions.

Furthermore, we define $N^S_n(t, d)$ to be the size of the largest intersection of two $t$-substitution balls of binary words of length $n$ and Hamming distance at least $d$. Mathematically speaking,

$$N^S_n(t, d) = \max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \{0,1\}^n \\ d_H(\boldsymbol{w}_1,\boldsymbol{w}_2) \geqslant d}} \left| B_{t,0,0}(\boldsymbol{w}_1) \cap B_{t,0,0}(\boldsymbol{w}_2) \right|.$$

Similarly, $N^D_n(t, d)$, $N^I_n(t, d)$ stands for the size of the largest intersection of two $t$-deletion, $t$-insertion balls of binary words of length $n$ and Levenshtein distance at least $d$, respectively, i.e.,

$$N^D_n(t, d) = \max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \{0,1\}^n \\ d_L(\boldsymbol{w},\boldsymbol{w}') \geqslant d}} \left| B_{0,t,0}(\boldsymbol{w}_1) \cap B_{0,t,0}(\boldsymbol{w}_2) \right|$$

and

$$N^I_n(t, d) = \max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \{0,1\}^n \\ d_L(\boldsymbol{w}_1,\boldsymbol{w}_2) \geqslant d}} \left| B_{0,0,t}(\boldsymbol{w}_1) \cap B_{0,0,t}(\boldsymbol{w}_2) \right|.$$

However, the more interesting value, especially in the light of DNA storage advances, is the size of the largest intersection of balls resulting from insertions, deletions, and substitutions. More specifically, calculating the following largest intersection size is of considerable importance.

$$N^*_n(t_1, t_2, t_3) = \max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \{0,1\}^n \\ \boldsymbol{w}_1 \neq \boldsymbol{w}_2}} \left| B^*_{t_1,t_2,t_3}(\boldsymbol{w}_1) \cap B^*_{t_1,t_2,t_3}(\boldsymbol{w}_2) \right|.$$

As stated earlier, the present paper tackles combinations of substitutions with either insertions and deletions. Therefore, we also define

$$N_n(t_1, t_2, t_3) = \max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2 \in \{0,1\}^n \\ \boldsymbol{w}_1 \neq \boldsymbol{w}_2}} \left| B_{t_1,t_2,t_3}(\boldsymbol{w}_1) \cap B_{t_1,t_2,t_3}(\boldsymbol{w}_2) \right|.$$

In addition, since

$$\left| B_{t_1,t_2,0}(\boldsymbol{w}) \right| = \sum_{i=0}^{t_2} \left| B_{t_1,i,0}(\boldsymbol{w}) \right|,$$

$$\left| B_{t_1,0,t_3}(\boldsymbol{w}) \right| = \sum_{i=0}^{t_3} \left| B_{t_1,0,i}(\boldsymbol{w}) \right|,$$

and

$$N^*_n(t_1, 0, t_3) = \sum_{i=0}^{t_3} N_n(t_1, 0, i),$$

in section III, IV, V we will first calculate the value of $\left| B_{t_1,1,0}(\boldsymbol{w}) \right|$, $\left| B_{1,0,1}(\boldsymbol{w}) \right|$, $N_n(1, 0, 1)$, and then derive the

value of $\left|B^*_{t_1,1,0}(\boldsymbol{w})\right|, \left|B^*_{1,0,1}(\boldsymbol{w})\right|, N^*_n(1,0,1)$, respectively. Lastly, we also emphasize that since studying the balls $B_{t,1,0}(\boldsymbol{w})$ and $B_{1,0,1}(\boldsymbol{w})$ leads to conclusions regarding the balls $B^*_{t,1,0}(\boldsymbol{w})$ and $B^*_{1,0,1}(\boldsymbol{w})$, then for convenience we will refer to $B_{t,1,0}(\boldsymbol{w})$ as the single-deletion single-substitution ball. Whenever the ball $B^*_{t,1,0}(\boldsymbol{w})$ is meant, it will be emphasized that at most one deletion case is discussed. Similarly, the term single-insertion single-substitution ball refers to the ball $B_{1,0,1}(\boldsymbol{w})$ and not $B^*_{1,0,1}(\boldsymbol{w})$.

Table I summarizes the main notations used throughout the paper. Moreover, for the convenience of the reader it provides additional notations that will be used only in specific sections. More details regarding each of the terms and the notations will be given in the section in which it is used.

### A. Previous Results Regarding the Reconstruction Problem With Substitutions

In [15], Levenshtein solved Problem 1 under the assumption that $B(\boldsymbol{w}) = B_{t,0,0}(\boldsymbol{w})$. Levenshtein also proved that this solution does not depend on the choice of the code, but on its minimum Hamming distance. More specifically, it was shown in [15] that

$$N^S_n(t,d) = \sum_{i=0}^{t-\lceil \frac{d}{2} \rceil} \binom{n-d}{i} \sum_{h=d-t+i}^{t-i} \binom{d}{h}.$$

In addition, another expression for the value of $N^S_n(t,d)$ was provided in [22]. Regarding Problems 2 and 3, significant advances were made in both of [15] and [22], and may be summarized as follows.

- Problem 3 was solved in [15] for $B(\boldsymbol{w}) = B_{t,0,0}(\boldsymbol{w})$ and $\mathcal{C} = \{0,1\}^n$. In [22], under specific assumptions, this solution was proved to hold for every binary code of length $n$ and minimum Hamming distance 2.
- Under certain assumptions, Problem 3 was solved in [22] for $B(\boldsymbol{w}) = B_{t,0,0}(\boldsymbol{w})$ and binary code $\mathcal{C}$ of length $n$ and minimum Hamming distance $d \in [3,4]$.
- Under certain assumptions, Problem 2 was solved in [22] for $B(\boldsymbol{w}) = B_{t,0,0}(\boldsymbol{w})$ and every binary code $\mathcal{C}$.

More details may be found in Section VI.

### B. Previous Results Regarding the Reconstruction Problem With Insertions and Deletions

Levenshtein presented in [16] the size of the $t$-insertion ball. More specifically, it was shown that for a word $\boldsymbol{w} \in \{0,1\}^n$, the size of the ball resulting from exactly $t$ insertions is given by

$$\left|B_{0,0,t}(\boldsymbol{w})\right| = \sum_{i=0}^{t} \binom{n+t}{i}.$$

On the other hand, calculating the size of the $t$-deletion ball, i.e., the ball that results from exactly $t$ deletions, is known to be a challenging problem by itself. Nevertheless, it was shown in [7] that the size of the largest $t$-deletion ball is given by

$$\max_{\boldsymbol{w} \in \{0,1\}^n} \left|B_{0,t,0}(\boldsymbol{w})\right| = \sum_{i=0}^{t} \binom{n-t}{i},$$

and is achieved by the alternating words.

Moreover, Levenshtein solved in [16] Problems 1 and 3 under the assumption that $\mathcal{C} = \{0,1\}^n$ for both of $B(\boldsymbol{w}) = B_{0,0,t}(\boldsymbol{w})$ and $B(\boldsymbol{w}) = B_{0,t,0}(\boldsymbol{w})$. That is, the values of $N^I_n(t,1)$ and $N^D_n(t,1)$ were determined in [16]. Later, the value of $N^I_n(t,d)$ for every $d$ and $t$ was found in [21], and the value of $N^D_n(t,2)$ for every $t$ was found in [5].

### C. Combinations of Insertions, Deletions, and Substitutions

An ambitious plan that one may aim to accomplish is to solve Problems 1, 2, and 3 for $\mathcal{C} = \{0,1\}^n$ and $B(\boldsymbol{w}) = B^*_{t_1,t_2,t_3}(\boldsymbol{w})$, and hence determine the value of $N^*_n(t_1,t_2,t_3)$ for every $t_1,t_2,t_3 \geqslant 0$. However, as explained earlier, before dealing with these problems, it is important to find the size of the ball $B^*_{t_1,t_2,t_3}(\boldsymbol{w})$, i.e., investigating Problem 4 for $B(\boldsymbol{w}) = B^*_{t_1,t_2,t_3}(\boldsymbol{w})$. We note that, to the best of our knowledge, finding the size of the ball $B^*_{t_1,t_2,t_3}(\boldsymbol{w})$ for all $t_1,t_2,t_3$ has not been studied before and is a challenging task by itself. The only solved case we are aware of is $(t_1,t_2,t_3) = (0,1,1)$, i.e., a single deletion and a single insertion [2], [20].

The rest of the paper takes the first steps towards solving Problems 1, 2, 3, and 4 for combinations of errors. More specifically, the following results are achieved.

- Solution to Problem 4 where $B(\boldsymbol{w}) = B_{t,1,0}(\boldsymbol{w})$, i.e., the size of the single-deletion $t$-substitution ball of any binary word is determined. This result also reveals the size of the ball $B^*_{t,1,0}(\boldsymbol{w})$ for every $t$ and $\boldsymbol{w}$.
- Solution to Problem 4 where $B(\boldsymbol{w}) = B_{1,0,1}(\boldsymbol{w})$, i.e., the size of the single-insertion single-substitution ball of any binary word is calculated. Therefore, the size of the ball $B^*_{1,0,1}(\boldsymbol{w})$ is also determined.
- Solution to Problem 1 where $B(\boldsymbol{w}) = B_{1,0,1}(\boldsymbol{w})$ and $\mathcal{C} = \{0,1\}^n$, i.e., the value of $N_n(1,0,1)$ is found. This result leads to value of $N^*_n(1,0,1)$ too.
- Solution to Problem 3 where $B(\boldsymbol{w}) = B_{1,0,1}(\boldsymbol{w})$ and $\mathcal{C} = \{0,1\}^n$, that is, an algorithm that receives words in a single-insertion single-substitution ball and returns the center word of the ball is provided. It will be shown that this algorithm achieves the minimum possible order of run-time complexity. Moreover, this algorithm is extended to the case of at most one insertion and at most one substitution.
- Solution to Problem 3 where $B(\boldsymbol{w}) = B_{t,0,0}(\boldsymbol{w})$, $\mathcal{C} = \{0,1\}^n$, and under the following assumptions.
  - $n$ is large enough with respect to $t$ and the minimum Hamming distance of $\mathcal{C}$.
  - There exists a complete decoder for the code $\mathcal{C}$.

Before we proceed to study the combinations of at most $t_1$ substitutions, exactly $t_2$ deletions, and $t_3$ insertions, i.e., the ball $B_{t_1,t_2,t_3}(\boldsymbol{w})$ for a binary word $\boldsymbol{w}$, we note that the order in which the deletions, insertions, and substitutions occur to generate the words of $B_{t_1,t_2,t_3}(\boldsymbol{w})$ does not matter. Thus, throughout the paper we will usually choose the order that simplifies our proofs.

## III. The Single-Deletion Multiple-Substitutions Ball Size

In this section, we study the size of the ball that results from at most $t$ substitutions and a single deletion, that is, the size

TABLE I
TERMINOLOGY AND NOTATIONS

| Notation | Description |
|---|---|
| | **Notations Used Throughout the Paper** |
| $[n, m]$ | $\{n, n+1, \ldots, m\}$. |
| $r(\boldsymbol{w})$ | Number of runs in $\boldsymbol{w}$. |
| $d_H(\boldsymbol{w}_1, \boldsymbol{w}_2)$ | Hamming distance between the words $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$. |
| $d_L(\boldsymbol{w}_1, \boldsymbol{w}_2)$ | Levenshtein distance between the words $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$, which is one half of the minimum number of insertions and deletions required to convert $\boldsymbol{w}_1$ to $\boldsymbol{w}_2$. |
| $B_{t_1,t_2,t_3}(\boldsymbol{w})$ | The set of words obtained from at most $t_1$ substitutions and exactly $t_2, t_3$ deletions, insertions in $\boldsymbol{w}$, respectively. |
| $B^*_{t_1,t_2,t_3}(\boldsymbol{w})$ | The set of words obtained from at most $t_1, t_2, t_3$, substitutions, deletions, insertions in $\boldsymbol{w}$, respectively. |
| $N_n^S(t, d)$ | $\max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2\in\{0,1\}^n \\ d_H(\boldsymbol{w}_1,\boldsymbol{w}_2)\geqslant d}} \lvert B_{t,0,0}(\boldsymbol{w}_1) \cap B_{t,0,0}(\boldsymbol{w}_2)\rvert$. |
| $N_n^D(t, d)$ | $\max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2\in\{0,1\}^n \\ d_L(\boldsymbol{w},\boldsymbol{w}')\geqslant d}} \lvert B_{0,t,0}(\boldsymbol{w}_1) \cap B_{0,t,0}(\boldsymbol{w}_2)\rvert$. |
| $N_n^I(t, d)$ | $\max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2\in\{0,1\}^n \\ d_L(\boldsymbol{w}_1,\boldsymbol{w}_2)\geqslant d}} \lvert B_{0,0,t}(\boldsymbol{w}_1) \cap B_{0,0,t}(\boldsymbol{w}_2)\rvert$. |
| $N_n(t_1, t_2, t_3)$ | $\max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2\in\{0,1\}^n \\ \boldsymbol{w}_1\neq\boldsymbol{w}_2}} \lvert B_{t_1,t_2,t_3}(\boldsymbol{w}_1) \cap B_{t_1,t_2,t_3}(\boldsymbol{w}_2)\rvert$. |
| $N_n^*(t_1, t_2, t_3)$ | $\max_{\substack{\boldsymbol{w}_1,\boldsymbol{w}_2\in\{0,1\}^n \\ \boldsymbol{w}_1\neq\boldsymbol{w}_2}} \left\lvert B^*_{t_1,t_2,t_3}(\boldsymbol{w}_1) \cap B^*_{t_1,t_2,t_3}(\boldsymbol{w}_2)\right\rvert$. |
| | **Notation Used in Section IV** |
| $\ell(\boldsymbol{w})$ | Runs-profile vector of word $\boldsymbol{w}$, which is a length-$r(\boldsymbol{w})$ vector that specifies the length of each of the $r(\boldsymbol{w})$ runs in the word $\boldsymbol{w}$. |
| | **Notations Used in Section V** |
| $B_{sub}(\boldsymbol{w})$ | The set of words received after exactly one substitution in the word $\boldsymbol{w}$. |
| $\boldsymbol{x} \searrow \widetilde{\boldsymbol{x}}$ in section $[k_1, k_2]$ | For words $\boldsymbol{x}, \widetilde{\boldsymbol{x}}$, the diagonal $\boldsymbol{x} \searrow \widetilde{\boldsymbol{x}}$ is maintained in section $[k_1, k_2]$, i.e., $x_k = \tilde{x}_{k+1}$ for every $k \in [k_1, k_2 - 1]$. |
| $\boldsymbol{x} \nearrow \widetilde{\boldsymbol{x}}$ in section $[k_1, k_2]$ | For words $\boldsymbol{x}, \widetilde{\boldsymbol{x}}$, the diagonal $\boldsymbol{x} \nearrow \widetilde{\boldsymbol{x}}$ is maintained in section $[k_1, k_2]$, i.e., $x_k = \tilde{x}_{k-1}$ for every $k \in [k_1 + 1, k_2]$. |
| | **Notations Used in Sections VI and VII** |
| $N_{t,d}$ | $N_n^S(t, d) + 1$ |
| $\mathcal{D}_{\mathcal{C}}$ | Complete decoder of code $\mathcal{C} \subseteq \{0,1\}^n$, i.e., a decoder that outputs a codeword for every input. However, can correct successfully at most $\frac{d-1}{2}$ errors, where $d$ is the minimum Hamming distance of $\mathcal{C}$. |
| $\mathrm{maj}_\tau(Y)$ | For a set $Y \subset \{0,1\}^n$ and threshold $\tau$, $\mathrm{maj}_\tau(Y)$ is a word $\boldsymbol{z}$ satisfying the following requirements. $z_i =?$ if the difference between the number of words in $Y$ with 1 in the $i$-th bit and the number of words with 0 in the same bit is less than or equal to $\tau$. Otherwise, $z_i$ is equal to the value with more occurrences in the $i$-th bit in $Y$. |

of $B_{t,1,0}(\boldsymbol{w})$ for all $\boldsymbol{w}$. For a word $\boldsymbol{w} \in \{0,1\}^n$, its number of runs is denoted by $r(\boldsymbol{w})$. The next lemma will be useful in proving our general result.

*Lemma 5:* Let $\boldsymbol{w} \in \{0,1\}^n$ be a word consisting of $r = r(\boldsymbol{w})$ runs. For $1 \leqslant i \leqslant r$, let $k_i$ be the starting index of the $i$-th run and $\boldsymbol{w}^i$ be the word received by deleting a bit from

the $i$-th run, so it holds that $B_{0,1,0}(\boldsymbol{w}) = \{\boldsymbol{w}^1, \boldsymbol{w}^2, \ldots, \boldsymbol{w}^r\}$. Denote by $D_{i,j}$ the set of indices in which $\boldsymbol{w}^i$ and $\boldsymbol{w}^j$ differ. For $1 \leqslant i < j \leqslant r$, the following properties hold.

1) $D_{i,j} = \{k_t - 1 : i + 1 \leqslant t \leqslant j\}$.
2) $d_H(\boldsymbol{w}^i, \boldsymbol{w}^j) = j - i$.
3) For $h$ such that $j < h \leqslant r$, it holds that $D_{j,h} \subseteq D_{i,h}$.

*Proof:* According to the definition of the words $\boldsymbol{w}^1, \ldots, \boldsymbol{w}^r$, the following equalities hold.

- $w^i_m = w^j_m = w_m$ for $m \in [1, k_i - 1]$.
- $w^i_m = w^j_m = w_{m+1}$ for $m \in [k_j, n-1]$.
- $w^i_m = w_{m+1}$ for $m \in [k_i, k_j - 1]$.
- $w^j_m = w_m$ for $m \in [k_i, k_j - 1]$.

Therefore, $D_{i,j} \subseteq [k_i, k_j - 1]$. Furthermore, $m \in D_{i,j}$ if and only if $w^i_m \neq w^j_m$, i.e., if and only if

$$w_{m+1} \neq w_m \text{ and } k_i \leqslant m \leqslant k_j - 1,$$

which holds only for $m \in \{k_t - 1 : i + 1 \leqslant t \leqslant j\}$. Hence, $|D_{i,j}| = j - i$ and this proves the first and the second claims of the lemma. Furthermore, for $j < h \leqslant r$ it is possible to conclude that $D_{i,h} = \{k_t - 1 : i + 1 \leqslant t \leqslant h\}$, $D_{j,h} = \{k_t - 1 : j + 1 \leqslant t \leqslant h\}$. Thus, $D_{j,h} \subseteq D_{i,h}$. ∎

We are now ready to prove the main result of this section, which is stated in the next theorem. However, we first emphasize that for $n \leqslant t + 1$ and $\boldsymbol{w} \in \{0,1\}^n$,

$$\left|B_{t,1,0}(\boldsymbol{w})\right| = 2^{n-1}$$

as every length-$(n-1)$ binary word may be received by single deletion and at most $t$ substitutions in $\boldsymbol{w}$. That is, all of the single-deletion $t$-substitution balls in case $n \leqslant t+1$ are of the same size and contain all of the length-$(n-1)$ binary words.

*Theorem 6:* The size of the single-deletion $t$-substitution ball of a word $\boldsymbol{w} \in \{0,1\}^n$ consisting of $r = r(\boldsymbol{w})$ runs is given by

$$\begin{aligned}|B_{t,1,0}(\boldsymbol{w})| &= \sum_{i=0}^{t} \binom{n-1}{i} \\ &+ \sum_{i=2}^{\min\{2,r\}} \binom{n-2}{t} + \sum_{i=3}^{\min\{3,r\}} \binom{n-3}{t} \\ &+ \sum_{i=4}^{r} \left( \binom{n-3}{t} - \sum_{j=2}^{\min\left\{t, \left\lfloor \frac{i-1}{2} \right\rfloor\right\}} C_{j-1} \binom{n-1-2j}{t-j} \right),\end{aligned}$$

where $C_i$ is the Catalan number $C_i = \frac{1}{i+1}\binom{2i}{i}$.

*Proof:* As stated in Section II, it is sufficient to consider the case where the single-deletion $t$-substitution ball consists of all of the words that can be generated by exactly one deletion followed by at most $t$ substitutions. Denote by $\boldsymbol{w}^i$ the word that results from a deletion in the $i$-th run in $\boldsymbol{w}$. Then $B_{0,1,0}(\boldsymbol{w}) = \{\boldsymbol{w}^1, \boldsymbol{w}^2, \ldots, \boldsymbol{w}^r\}$ and

$$B_{t,1,0}(\boldsymbol{w}) = \bigcup_{i=1}^{r} B_{t,0,0}(\boldsymbol{w}^i).$$

Hence, $B_{t,1,0}(\boldsymbol{w})$ consists of the union of the following mutually disjoint sets

$$S_i = B_{t,0,0}(\boldsymbol{w}^i) \setminus \bigcup_{j=1}^{i-1} B_{t,0,0}(\boldsymbol{w}^j),$$

where $i \in [1, r]$. Thus, $|B_{t,1,0}(\boldsymbol{w})| = \sum_{i=1}^{r} |S_i|$ and in order to find the value of $|B_{t,1,0}(\boldsymbol{w})|$, it is enough to find the size of $S_i, i \in [1, r]$. Based on Lemma 5, the intersection $B_{t,0,0}(\boldsymbol{w}^i) \cap B_{t,0,0}(\boldsymbol{w}^j)$ is empty if and only if $|i - j| > 2t$. Thus,

$$S_i = B_{t,0,0}(\boldsymbol{w}^i) \setminus \bigcup_{j=\max\{1,i-2t\}}^{i-1} B_{t,0,0}(\boldsymbol{w}^j).$$

Observe that for each word $\boldsymbol{w}^i, i \geqslant 2t+1$, there are exactly $2t$ previous words to consider, $\boldsymbol{w}^{i-2t}, \ldots, \boldsymbol{w}^{i-1}$. However, for words $\boldsymbol{w}^i, i \leqslant 2t$ there are only $i-1$ previous words, $\boldsymbol{w}^1, \boldsymbol{w}^2, \ldots, \boldsymbol{w}^{i-1}$. Furthermore, it can be seen in the next steps that calculating $|S_i|$ does not depend on the specific value of $i$, i.e., $|S_i| = |S_j|$ for $i, j \in [2t+1, r]$. Considering these claims we will show how to find $|S_i|$ for $i \in [2t+1, r]$ and deduce the values of $|S_i|, i \in [1, 2t]$ using the same approach. More accurately, we will find that

$$|S_i| = \begin{cases} \sum_{h=0}^{t} \binom{n-1}{h}, & i = 1 \\ \binom{n-2}{t}, & i = 2 \\ \binom{n-3}{t}, & i = 3 \\ \binom{n-3}{t} - \sum_{j=2}^{\min\left\{t, \left\lfloor \frac{i-1}{2} \right\rfloor\right\}} C_{j-1} \binom{n-1-2j}{t-j}, & i \in [4, r] \end{cases}$$

which proves the theorem's statement.

Given a word $\boldsymbol{w}^i, i \in [2t+1, r]$, in order to find the set $S_i$, it is enough to exclude from $B_{t,0,0}(\boldsymbol{w}^i)$ the following disjoint sets

$$T_j = \left( B_{t,0,0}(\boldsymbol{w}^i) \cap B_{t,0,0}(\boldsymbol{w}^{i-j}) \right) \setminus \bigcup_{h=1}^{j-1} B_{t,0,0}(\boldsymbol{w}^{i-h}),$$

where $j \in [1, 2t]$. In other words, first $B_{t,0,0}(\boldsymbol{w}^i) \cap B_{t,0,0}(\boldsymbol{w}^{i-1})$ is excluded from the set $B_{t,0,0}(\boldsymbol{w}^i)$. Then, words in $B_{t,0,0}(\boldsymbol{w}^i) \cap B_{t,0,0}(\boldsymbol{w}^{i-2})$ that were not excluded before are excluded, i.e., $\left(B_{t,0,0}(\boldsymbol{w}^i) \cap B_{t,0,0}(\boldsymbol{w}^{i-2})\right) \setminus B_{t,0,0}(\boldsymbol{w}^{i-1})$, etc., until the set $\left(B_{t,0,0}(\boldsymbol{w}^i) \cap B_{t,0,0}(\boldsymbol{w}^{i-2t})\right) \setminus \bigcup_{j=1}^{2t-1} B_{t,0,0}(\boldsymbol{w}^{i-j})$ is excluded at the end.

Hence,

$$S_i = B_{t,0,0}(\boldsymbol{w}^i) \setminus \bigcup_{j=1}^{2t} T_j,$$

and $|S_i| = |B_{t,0,0}(\boldsymbol{w}^i)| - \sum_{j=1}^{2t} |T_j|$. Remember that $|B_{t,0,0}(\boldsymbol{w}^i)| = \sum_{h=0}^{t} \binom{n-1}{h}$. In the next steps the following equality is proved .

$$|T_j| = \begin{cases} \binom{n-2}{t-1} + \sum_{h=0}^{t-1} \binom{n-1}{h}, & j = 1 \\ \binom{n-3}{t-1}, & j = 2 \\ 0, & j \text{ odd number in } [3, 2t] \\ C_{\frac{j}{2}-1} \binom{n-1-j}{t-\frac{j}{2}}, & j \text{ even number in } [3, 2t]. \end{cases}$$

This leads to the claimed values of $|S_i|, i \in [1, r]$.

In order to find the size of $T_j, j \in [1, 2t]$, we begin with noting that based on Lemma 5, there exists a set of $2t$ positions $\{i_1, i_2, \ldots, i_{2t}\}$ such that, for $j \in [1, 2t]$, $\boldsymbol{w}^i$ differs from $\boldsymbol{w}^{i-j}$ in positions $i_1, i_2, \ldots, i_j$.

Since $\boldsymbol{w}^i$ and $\boldsymbol{w}^{i-1}$ differ only in one position, then $T_1$ consists of the following disjoint sets of words.

1) Words that result from at most $t-1$ substitutions in $\boldsymbol{w}^i$.
2) Words that result from $t$ substitutions in $\boldsymbol{w}^i$ such that one of them is in position $i_1$.

Hence,

$$|T_1| = \binom{n-1-1}{t-1} + \sum_{j=0}^{t-1} \binom{n-1}{j}.$$

Regarding $\boldsymbol{w}^{i-2}$, since it differs from $\boldsymbol{w}^i$ in positions $i_1$ and $i_2$ and since we do not consider words that have been excluded, then $T_2$ consists of words that result from exactly $t$ substitutions in $\boldsymbol{w}^i$ such that one of them is in position $i_2$ and none of them is in position $i_1$. Hence, $|T_2| = \binom{n-1-2}{t-1}$.

The value $|T_j|, j \in [3, 2t]$ is found using the following three claims, where the proofs of the first and the third one are presented in Appendix A.

*Claim 1:* The set $T_j$, $j \in [3, 2t]$ consists only of the words that satisfy all of the following requirements.

1) Result from exactly $t$ substitutions in $\boldsymbol{w}^i$.
2) Identical to $\boldsymbol{w}^i$ in positions $i_1, i_2$.
3) Result from at least $\left\lceil \frac{j}{2} \right\rceil$ substitutions in positions $i_1, i_2, \ldots, i_j$ in $\boldsymbol{w}^i$.
4) Result from at most $\left\lceil \frac{h}{2} \right\rceil - 1$ substitutions in positions $i_1, i_2, \ldots, i_h$ in $\boldsymbol{w}^i$, for every $3 \leqslant h < j$.

*Claim 2:* For $j \in [3, 2t]$ odd, $|T_j| = 0$.

*Proof:* It is enough to prove that for $j \in [3, 2t]$ odd, there is no word that satisfies all of the requirements stated in Claim 1. Assume to the contrary that there exits such a word. Based on the fourth requirement, such a word has at most $\left\lceil \frac{j-1}{2} \right\rceil - 1 = \left\lfloor \frac{j}{2} \right\rfloor - 1$ substitutions in positions $i_1, i_2, \ldots, i_{j-1}$. Hence, it has at most $\left\lfloor \frac{j}{2} \right\rfloor$ substitutions in all of the positions $i_1, i_2, \ldots, i_j$, which contradicts the third requirement. ∎

*Claim 3:* For $j \in [3, 2t]$ even, $|T_j| = C_{\frac{j}{2}-1} \binom{n-1-j}{t-\frac{j}{2}}$.

Lastly, note that using the same approach, it is possible to get that

$$|S_2| = \binom{n-2}{t}, |S_3| = \binom{n-3}{t},$$

and for $4 \leqslant i \leqslant 2t$,

$$|S_i| = \binom{n-3}{t} - \sum_{j=2}^{\left\lfloor \frac{i-1}{2} \right\rfloor} C_{j-1} \binom{n-1-2j}{t-j},$$

which concludes the proof of the theorem. ∎

In the next example we demonstrate the proof of Theorem 6 in order to find the size of the ball $B_{t,1,0}(\boldsymbol{x})$.

*Example 1:* Assume $t = 4$ and $\boldsymbol{w}$ as presented in the following table, consists of 9 runs, each of length 2. As stated earlier, the single-deletion $t$-substitution ball can be received by first deleting some bit and then applying at most $t$ substitutions. Thus, we begin with considering the single deletion ball $\{\boldsymbol{w}^1, \boldsymbol{w}^2, \ldots, \boldsymbol{w}^9\}$.

| Index | $\boldsymbol{w}$ | $\boldsymbol{w}^1$ | $\boldsymbol{w}^2$ | $\boldsymbol{w}^3$ | $\boldsymbol{w}^4$ | $\boldsymbol{w}^5$ | $\boldsymbol{w}^6$ | $\boldsymbol{w}^7$ | $\boldsymbol{w}^8$ | $\boldsymbol{w}^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 4 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 8 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 11 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 12 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| 15 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 16 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 18 | 0 | | | | | | | | | |

The simplest algorithm that can generate $B_{t,1,0}(\boldsymbol{w})$ goes iteratively over the words $\boldsymbol{w}^i, 1 \leqslant i \leqslant 9$ and each time adds $B_{t,0,0}(\boldsymbol{w}^i)$ to a set $S$. At the end we will have $S = B_{t,1,0}(\boldsymbol{w})$. However, since we do not want to add the same word to $S$ multiple times, we aim to find for every $i \in [1, r]$, the value of $|B_{t,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=1}^{i-1} B_{t,0,0}(\boldsymbol{w}^j)|$ and subtract it from $|B_{t,0,0}(\boldsymbol{w}^i)|$. As specified in the proof of Theorem 6, for each $i \in [1, r]$, we need to consider at most eight previous words (if they exist), hence, we focus on $\boldsymbol{w}^9$ and find the size of the set

$$B_{t,0,0}(\boldsymbol{w}^9) \cap \bigcup_{j=1}^{8} B_{t,0,0}(\boldsymbol{w}^j).$$

We accomplish this task in the following steps.

1) First, the sets $B_{t,0,0}(\boldsymbol{w}^9) \cap B_{t,0,0}(\boldsymbol{w}^8)$ and $B_{t,0,0}(\boldsymbol{w}^9) \cap B_{t,0,0}(\boldsymbol{w}^7)$ are excluded from $B_{t,0,0}(\boldsymbol{w}^9)$. It can been seen that the union of these two sets consists of all of the words of Hamming distance at most 3 from $\boldsymbol{y}_9$ , and all of the words of Hamming distance 4 from $\boldsymbol{w}^9$ that differ from $\boldsymbol{w}^9$ in at least one of the 16-th and 14-th positions.
2) Next, $B_{t,0,0}(\boldsymbol{w}^9) \cap B_{t,0,0}(\boldsymbol{w}^6)$ is excluded from $B_{t,0,0}(\boldsymbol{w}^9)$. It can be seen that based on the first step, no word should be excluded now.
3) $B_{t,0,0}(\boldsymbol{w}^9) \cap B_{t,0,0}(\boldsymbol{w}^5)$ is excluded from $B_{t,0,0}(\boldsymbol{w}^9)$. Based on the previous steps, only words that result from 4 substitutions such that two of them are in positions 10, 12 and none are in positions 14, 16 are excluded at this step.
4) Regarding $B_{t,0,0}(\boldsymbol{w}^9) \cap B_{t,0,0}(\boldsymbol{w}^4)$, based on Claim 2 and the previous steps no word should be excluded.
5) $B_{t,0,0}(\boldsymbol{w}^9) \cap B_{t,0,0}(\boldsymbol{w}^3)$ is excluded from $B_{t,0,0}(\boldsymbol{w}^9)$. Hence, words in which there are exactly two substitutions in positions 6, 8, one in positions 10, 12 and none in positions 14, 16 are excluded at this step.
6) Regarding $B_{t,0,0}(\boldsymbol{w}^9) \cap B_{t,0,0}(\boldsymbol{w}^2)$, based on Claim 2 and the previous steps no word should be excluded.
7) Exclude words in $B_{t,0,0}(\boldsymbol{w}^9) \cap B_{t,0,0}(\boldsymbol{w}^1)$ from $B_{t,0,0}(\boldsymbol{w}^9)$. We observe that $\boldsymbol{w}^1$ differs from $\boldsymbol{w}^9$ in positions 2, 4, 6, 8, 10, 12, 14 , and 16. According to the previous

steps, at this step we aim to exclude words satisfying all of the following requirements.

- Generated by exactly 4 substitutions in $\boldsymbol{y}_9$.
- Identical to $\boldsymbol{y}_9$ in positions 14 and 16.
- Differ from $\boldsymbol{y}_9$ in at most one position among 10, 12, 14, 16.
- Differ from $\boldsymbol{y}_9$ in at most two positions among 6, 8, 10, 12, 14, 16.
- Differ from $\boldsymbol{y}_9$ in at least four positions among 2, 4, 6, 8, 10, 12, 14, 16.

In other words, we need to find the number of possibilities for applying substitutions in $\boldsymbol{y}_9$ in positions 2, 4, 6, 8, 10, 12 such that all of the following requirements hold.

- At most one substitution in 10, 12.
- At most two substitutions in 6, 8, 10, 12.
- Exactly four substitutions in 2, 4, 6, 8, 10, 12.
- Two substitutions in 2, 4.

Here we note that if we reinterpret a substitution as a $''$)$''$ and no substitution as a $''$($''$, then we find that the number of the possibilities we want to count is equal to the number of length-4 expressions among the alphabet $\{(,)\}$ containing 2 pairs of parentheses such that in every prefix of even length the number of $''$)$''$ is less than or equal to the number of $''$($''$. The number of such expressions is equal to $C_3$.

A direct result of Theorem 6 is given in the next corollary, which its complete proof can also be found in [1].

*Corollary 7:* The single-deletion single-substitution ball size of $\boldsymbol{w} \in \{0,1\}^n$ is

$$|B_{1,1,0}(\boldsymbol{w})| = (n-3)r(\boldsymbol{w}) + 4,$$

for $r(\boldsymbol{w}) \geqslant 2$ and $|B_{1,1,0}(\boldsymbol{w})| = n$ if $r(\boldsymbol{w}) = 1$.

Based on Corollary 7, we note that, for $n \geqslant 4$, the maximum size of the single-substitution single-deletion ball is $n^2-3n+4$ and is received only for the alternating word, while the minimum size is $n$ and is received only for a single run word. The following corollary extends this observation to every single-deletion multiple-substitution ball. The proof follows directly from the result of Theorem 6, as it can be seen that the larger $r(\boldsymbol{w})$ is, the larger $\left|B_{t,1,0}(\boldsymbol{w})\right|$ becomes.

*Corollary 8:* The largest single-deletion $t$-substitution ball is achieved by the alternating words and is of size

$$\sum_{i=0}^{t}\binom{n-1}{i}+\binom{n-2}{t}+\binom{n-3}{t} + \sum_{i=4}^{n}\left(\binom{n-3}{t} - \sum_{j=2}^{\min\left\{t,\left\lfloor\frac{i-1}{2}\right\rfloor\right\}} C_{j-1}\binom{n-1-2j}{t-j}\right).$$

Furthermore, the smallest single-deletion $t$-substitution ball is achieved by the single run words and is of size

$$\sum_{i=0}^{t}\binom{n-1}{i}.$$

For $n \leqslant t+1$, all of the single-deletion $t$-substitution balls are of size $2^{n-1}$. However, if $n$ satisfies the inequality

$$\binom{n-3}{t} - \sum_{j=2}^{t} C_{j-1}\binom{n-1-2j}{t-j} > 0,$$

then the largest, smallest ball is achieved only by the alternating, single run words, respectively.

Lastly, the following corollary extends the ball size result to the case of at most one deletion and at most $t$ substitutions.

*Corollary 9:* Let $\boldsymbol{w}$ be a length-$n$ binary word consisting of $r = r(\boldsymbol{w})$ runs. Then the number of words obtained by at most one deletion and at most $t$ substitutions in $\boldsymbol{w}$ is given by

$$|B^{*}_{t,1,0}(\boldsymbol{w})| = \sum_{i=0}^{t}\binom{n}{i} + \sum_{i=0}^{t}\binom{n-1}{i} + \sum_{i=2}^{\min\{2,r\}}\binom{n-2}{t} + \sum_{i=3}^{\min\{3,r\}}\binom{n-3}{t} + \sum_{i=4}^{r}\left(\binom{n-3}{t} - \sum_{j=2}^{\min\left\{t,\left\lfloor\frac{i-1}{2}\right\rfloor\right\}} C_{j-1}\binom{n-1-2j}{t-j}\right).$$

## IV. The Single-Insertion Single-Substitution Ball Size

For a word $\boldsymbol{w} \in \{0,1\}^n$ consisting of $r(\boldsymbol{w})$ runs, we define its *runs-profile vector* to be a length-$r(\boldsymbol{w})$ vector $\ell(\boldsymbol{w}) = (\ell_1, \ldots, \ell_{r(\boldsymbol{w})})$, which specifies the length of each of the $r(\boldsymbol{w})$ runs in $\boldsymbol{w}$. The runs-profile vector determines the word $\boldsymbol{w}$ up to its complement. In this section, we find the size of the single-insertion single-substitution ball, i.e., the ball that results from a single insertion and at most a single substitution. This result is proved in the next theorem.

*Theorem 10:* Let $\boldsymbol{w} \in \{0,1\}^n$ be a word with runs-profile vector $\ell(\boldsymbol{w}) = (\ell_1, \ell_2, \ldots, \ell_{r(\boldsymbol{w})})$. Then, the size of its single-insertion single-substitution ball is given by

$$|B_{1,0,1}(\boldsymbol{w})| = (n+2)^2 - 2 - \sum_{i=1}^{r(\boldsymbol{w})}\frac{\ell_i(\ell_i+5)}{2}.$$

*Proof:* Let $\boldsymbol{w}^i, i \in [1,n]$ be the word generated by inserting $\overline{w}_i$ after $w_i$ in $\boldsymbol{w}$. Moreover, let $\boldsymbol{w}^{-1}, \boldsymbol{w}^0$ be the word received by inserting $w_1, \overline{w}_1$ at the beginning of $\boldsymbol{w}$, respectively. It holds that $B_{0,0,1}(\boldsymbol{w}) = \{\boldsymbol{w}^i : i \in [-1,n]\}$. Hence, the single-insertion single-substitution ball is given by

$$B_{1,0,1}(\boldsymbol{w}) = \bigcup_{\boldsymbol{z}\in B_{0,0,1}(\boldsymbol{w})} B_{1,0,0}(\boldsymbol{z}) = \bigcup_{i=-1}^{n} B_{1,0,0}(\boldsymbol{w}^i),$$

which can also be expressed by the following union of disjoint sets

$$B_{1,0,1}(\boldsymbol{w}) = \bigcup_{i=-1}^{n}\left(B_{1,0,0}(\boldsymbol{w}^i) \setminus \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j)\right).$$

Thus, $|B_{1,0,1}(\boldsymbol{w})|$ can be obtained by the following sum

$$|B_{1,0,1}(\boldsymbol{w})| = \sum_{i=-1}^{n} \left| B_{1,0,0}(\boldsymbol{w}^i) \setminus \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j) \right| = \sum_{i=-1}^{n} |B_{1,0,0}(\boldsymbol{w}^i)| - \left| B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j) \right|.$$

Since $|B_{1,0,0}(z)| = n+2$ for any word $z \in B_{0,0,1}(\boldsymbol{w})$ and since $B_{1,0,0}(\boldsymbol{w}^{-1}) \cap B_{1,0,0}(\boldsymbol{w}^0) = \{w_1\boldsymbol{w}, \overline{w}_1\boldsymbol{w}\}$, then

$$|B_{1,0,1}(\boldsymbol{w})| = n+2+n+2-2 + \sum_{i=1}^{n} n+2 - \left| B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j) \right| = (n+2)^2 - 2 - \sum_{i=1}^{n} \left| B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j) \right|.$$

Hence, it is enough to prove that

$$\sum_{i=1}^{n} \left| B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j) \right| = \sum_{i=1}^{r(\boldsymbol{w})} \frac{\ell_i(\ell_i+5)}{2}.$$

In order to establish the last equation, we aim to find the size of each of the following $n$ intersections

$$B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j),$$

where $i \in [1, n]$. Let $0 < j_1 < j_2 \leqslant n$ and assume that the $j_2$-th bit falls in the $h$-th run in $\boldsymbol{w}$. In the following claims we consider all of the possibilities for $j_1$ and for each one we find the value of $|B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2})|$. Lastly, we consider the words $\boldsymbol{w}^{-1}, \boldsymbol{w}^0$. The following property is well known by Levenshtein [15] and will be used in our proof.

*Proposition 11:* $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) = \emptyset$ if and only if $d_H(\boldsymbol{w}^{j_1}, \boldsymbol{w}^{j_2}) > 2$.

Next, we present several claims that will be used in the proof of the theorem. However, we leave the proofs of some of them to Appendix B.

*Claim 4:* The words $\boldsymbol{w}^{j_1}, \boldsymbol{w}^{j_2}$ differ only in the following positions and under the following conditions.

- The $(j_1+1)$-st position, only if $w_{j_1} = w_{j_1+1}$.
- Position $i \in [j_1+2, j_2]$, only if $w_i \neq w_{i-1}$, i.e., a new run begins in the $i$-th position in $\boldsymbol{w}$.
- The $(j_2+1)$-st position.

*Proof:* Observe that the following equalities hold .

- $w_i^{j_1} = w_i, w_i^{j_2} = w_i$, where $1 \leqslant i \leqslant j_1$.
- $w_{j_1+1}^{j_1} = \overline{w}_{j_1}, w_{j_1+1}^{j_2} = w_{j_1+1}$.
- $w_i^{j_1} = w_{i-1}, w_i^{j_2} = w_i$, where $j_1+2 \leqslant i \leqslant j_2$.
- $w_{j_2+1}^{j_1} = w_{j_2}, w_{j_2+1}^{j_2} = \overline{w}_{j_2}$.
- $w_i^{j_1} = w_{i-1}, w_i^{j_2} = w_{i-1}$, where $j_2+2 \leqslant i \leqslant n+1$.

Hence, $\boldsymbol{w}^{j_1}, \boldsymbol{w}^{j_2}$ differ in the $(j_2+1)$-st position and in positions in $[j_1+1, j_2]$ under the conditions stated in the claim. ∎

*Claim 5:* If the $j_1$-th bit falls in the $(h-3)$-rd run or a previous one, then $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) = \emptyset$.

*Claim 6:* Assume the $j_1$-th bit falls in the $(h-2)$-nd or $(h-1)$-st run. Then, $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) \neq \emptyset$ if and only if the $j_1$-th bit is the last one in its run.

*Claim 7:* If the $j_1$-th bit falls in the $h$-th run, then $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) \neq \emptyset$.

*Claim 8:* For $i \geqslant 1$, the following statements hold.

- $B_{1,0,0}(\boldsymbol{w}^{-1}) \cap B_{1,0,0}(\boldsymbol{w}^i) \neq \emptyset$ if and only if the $i$-th bit falls in the first or the second run.
- $B_{1,0,0}(\boldsymbol{w}^0) \cap B_{1,0,0}(\boldsymbol{w}^i) \neq \emptyset$ if and only if the $i$-th bit falls in the first run.

*Claim 9:* Assume $B_{1,0,0}(\boldsymbol{w}^{i_1}) \cap B_{1,0,0}(\boldsymbol{w}^{i_2}) \neq \emptyset$ where $-1 \leqslant i_1 < i_2 \leqslant n$ and the $i_2$-th bit in $\boldsymbol{w}$ falls in the $h'$-th run, then $\boldsymbol{w}^{i_1}$ can be received by some insertion to the $h'$-th run, including insertion of $w_{i_2}$ or $\overline{w}_{i_2}$ at the beginning of the $h'$-th run.

To summarize, given a word $\boldsymbol{w}^i$, where $-1 \leqslant i \leqslant n$ and the $i$-th bit falls in the $h'$-th run, assume that the $i$-th bit is the $j$-th one in the $h'$-th run. Then the words $\boldsymbol{w}^{i'}, -1 \leqslant i' < i$ for which $B_{1,0,0}(\boldsymbol{w}^{i'}) \cap B_{1,0,0}(\boldsymbol{w}^i) \neq \emptyset$ are given by the following list.

- Word $\boldsymbol{w}'$ which results from an insertion of $\overline{w}_i$ at the beginning of the $h'$-th run.
- Word $\boldsymbol{w}''$ which results from an insertion of $w_i$ at the beginning of the $h'$-th run.
- All words $\boldsymbol{w}^{i'}$ where the $i'$-th bit falls in the $h'$-th run in $\boldsymbol{w}$. Note that there are $j-1$ such words.

It holds that $\boldsymbol{w}''$ differs from $\boldsymbol{w}^i$ only in the $(i+1)$-st position. Hence, $B_{1,0,0}(\boldsymbol{w}'') \cap B_{1,0,0}(\boldsymbol{w}^i) = \{\boldsymbol{w}'', \boldsymbol{w}^i\}$. Note that $\boldsymbol{w}''$ can be received by substituting the $(i+1)$-st bit in $\boldsymbol{w}^i$. Furthermore, $\boldsymbol{w}'$ differs from $\boldsymbol{w}^i$ in positions $i-j+1$ and $i+1$. Thus, $B_{1,0,0}(\boldsymbol{w}') \cap B_{1,0,0}(\boldsymbol{w}^i)$ consists of two words. The first one is received by substituting the $(i-j+1)$-st bit in $\boldsymbol{w}^i$ while the second is received by substituting the $(i+1)$-st bit in $\boldsymbol{w}^i$. Regarding $\boldsymbol{w}^{i'}$ where the $i'$-th bit falls in the $h'$-th run, it holds that $\boldsymbol{w}^{i'}$ differs from $\boldsymbol{w}^i$ in positions $i'+1$ and $i+1$. Hence, $B_{1,0,0}(\boldsymbol{w}^{i'}) \cap B_{1,0,0}(\boldsymbol{w}^i)$ consists of the words received by substituting one of the $(i'+1)$-st and $(i+1)$-st positions in $\boldsymbol{w}^i$.

By considering all of these intersections, it is possible to conclude that $B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{k=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^k)$ consists of the word $\boldsymbol{w}^i$ itself and any word received by substituting any of the bits in section $[i-j+1, i+1]$ in $\boldsymbol{w}^i$. Hence,

$$\left| B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{k=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^k) \right| = 1+i+1-(i-j+1)+1 = j+2.$$

Furthermore, let $a_k$ be the position in which the $k$-th run begins, i.e., $a_k = 1 + \sum_{i=1}^{k-1} \ell_i$. It holds that

$$\sum_{i=1}^{n} \left| B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j) \right| = \sum_{k=1}^{r(\boldsymbol{w})} \sum_{m=a_k}^{a_k+\ell_k-1} \left| B_{1,0,0}(\boldsymbol{w}^m) \cap \bigcup_{j=-1}^{m-1} B_{1,0,0}(\boldsymbol{w}^j) \right| = \sum_{k=1}^{r(\boldsymbol{w})} \sum_{m=a_k}^{a_k+\ell_k-1} m - a_k + 3 = \sum_{k=1}^{r(\boldsymbol{w})} \sum_{m=0}^{\ell_k-1} m+3$$

$$= \sum_{k=1}^{r(\boldsymbol{w})} \sum_{m=3}^{3+\ell_k-1} m = \sum_{k=1}^{r(\boldsymbol{w})} \frac{\ell_k(3+3+\ell_k-1)}{2}$$
$$= \sum_{k=1}^{r(\boldsymbol{w})} \frac{\ell_k(\ell_k+5)}{2},$$

which proves the lemma. ∎

Next we present an example of the calculation of the single-insertion single-substitution ball.

*Example 2:* Let $n = 8$ and $\boldsymbol{w}$ be the word presented in the following table with its single-insertion ball $\{\boldsymbol{w}^i : i \in [-1, n]\}$.

| Index | $\boldsymbol{w}$ | $\boldsymbol{w}^{-1}$ | $\boldsymbol{w}^0$ | $\boldsymbol{w}^1$ | $\boldsymbol{w}^2$ | $\boldsymbol{w}^3$ | $\boldsymbol{w}^4$ | $\boldsymbol{w}^5$ | $\boldsymbol{w}^6$ | $\boldsymbol{w}^7$ | $\boldsymbol{w}^8$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 4 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 5 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 7 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 8 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| 9 | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

We observe that, for any $i \in [-1, n]$, $|B_{1,0,0}(\boldsymbol{w}^i)| = n+2$ and that we have $n+2$ words in the single-insertion ball of $\boldsymbol{w}$. Moreover, the intersection $B_{1,0,0}(\boldsymbol{w}^{-1}) \cap B_{1,0,0}(\boldsymbol{w}^0)$ consists of the two words $\boldsymbol{w}^{-1}, \boldsymbol{w}^0$. Hence, $|B_{1,0,0}(\boldsymbol{w}^{-1}) \cup B_{1,0,0}(\boldsymbol{w}^0)| = 2(n+2) - 2 = 18$. Furthermore, all of the following claims hold.

- $\boldsymbol{w}^{-1}, \boldsymbol{w}^0, \boldsymbol{w}^1$ and $\boldsymbol{w}^2$ are words that result from insertions to the first run.
- $\boldsymbol{w}^{-1}, \boldsymbol{w}^2, \boldsymbol{w}^3$ and $\boldsymbol{w}^4$ are words that result from insertions to the second run.
- $\boldsymbol{w}^2, \boldsymbol{w}^4, \boldsymbol{w}^5$ and $\boldsymbol{w}^6$ are words that result from insertions to the third run.
- $\boldsymbol{w}^4, \boldsymbol{w}^6, \boldsymbol{w}^7$ and $\boldsymbol{w}^8$ are words that result from insertions to the fourth run.
- $B_{1,0,0}(\boldsymbol{w}^1) \cap \left(B_{1,0,0}(\boldsymbol{w}^{-1}) \cup B_{1,0,0}(\boldsymbol{w}^0)\right)$ consists of $\boldsymbol{w}^1$ and the two words that result from substituting the first or the second bit in $\boldsymbol{w}^1$. Similarly, $B_{1,0,0}(\boldsymbol{w}^2) \cap \left(B_{1,0,0}(\boldsymbol{w}^{-1}) \cup B_{1,0,0}(\boldsymbol{w}^0) \cup B_{1,0,0}(\boldsymbol{w}^1)\right)$ consists of the word $\boldsymbol{w}^2$ itself and the words that result from substituting one of the first, second and third bits in $\boldsymbol{w}^2$.
- The words $\boldsymbol{w}^3, \boldsymbol{w}^4$ are of Hamming distance 3 from the words $\boldsymbol{w}^0, \boldsymbol{w}^1$. Hence, when finding the intersection $B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j)$ for $i \in \{3, 4\}$ we need to consider only the words $\boldsymbol{w}^{-1}, \boldsymbol{w}^2$, and additionally $\boldsymbol{w}^3$ for $i = 4$. More specifically, the size of the intersection $B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j)$ is 3, 4 for $i = 3, 4$, respectively.
- The size of the intersection $B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j)$ is 3, 4 for $i = 5, 6$, respectively.
- The size of the intersection $B_{1,0,0}(\boldsymbol{w}^i) \cap \bigcup_{j=-1}^{i-1} B_{1,0,0}(\boldsymbol{w}^j)$ is 3, 4 for $i = 7, 8$, respectively.

Hence,

$$|B_{1,0,0}(\boldsymbol{w})| = 2(n+2) - 2 + \sum_{i=1}^{r(\boldsymbol{w})} (n+2-3+n+2-4)$$
$$= (n+2)^2 - 2 - 4(3+4) = 70.$$

Lastly, we note that the maximum size of the single-insertion single-substitution ball is $n^2+n+2$ and is received for the alternating word while the minimum size is $\frac{n^2}{2} + \frac{3n}{2} + 2$ and is received for a single run word. Furthermore, the following corollary extends the size of the single-insertion single-substitution ball to the case of at most one insertion and at most one substitution.

*Corollary 12:* Let $\boldsymbol{w} \in \{0,1\}^n$ be a word with runs-profile vector $\ell(\boldsymbol{w}) = (\ell_1, \ell_2, \ldots, \ell_{r(\boldsymbol{w})})$. Then, the number of words obtained by at most one insertion and at most one substitution in $\boldsymbol{w}$ is given by

$$\left|B^*_{1,0,1}(\boldsymbol{w})\right| = n+1+(n+2)^2 - 2 - \sum_{i=1}^{r(\boldsymbol{w})} \frac{\ell_i(\ell_i+5)}{2}.$$

## V. Reconstruction From a Single Substitution and a Single Insertion

In this section we make first steps towards studying Problem 1 for combinations of errors. More specifically, we focus on the case of single-insertion single-substitution and show that

$$N_n(1,0,1) = \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n.$$

Here we emphasize that for words of length $n \leqslant 2$, the largest single-insertion single-substitution intersection can be all of the space, for example $B_{1,0,1}(0) \cap B_{1,0,1}(1) = \{0,1\}^2$ and $B_{1,0,1}(10) \cap B_{1,0,1}(01) = \{0,1\}^3$. Moreover, for $n \geqslant 10$, $\left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 1 \leqslant \frac{n^2}{2} + \frac{n}{2} + 2$, i.e., $N_n(1,0,1)+1$ is smaller than or equal to the smallest single-insertion single-substitution ball and for $n \geqslant 3$, $\left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 1 \leqslant n^2+n+2$, i.e., $N_n(1,0,1)+1$ is smaller than or equal to the largest single-insertion single-substitution ball. First, we prove the lower bound on $N_n(1,0,1)$ in Lemma 15 by presenting two words for which the size of their balls' intersection is $\left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n$. Next, we prove the upper bound by presenting a decoder that gets $\left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 1$ different words in $B_{1,0,1}(\boldsymbol{w})$ where $\boldsymbol{w} \in \{0,1\}^n$, $n \geqslant 3$, and returns $\boldsymbol{w}$. We will also prove that this is an optimal decoder as its time complexity is $\Theta(n^3)$. Finally, we find it important to note that while the largest intersection in the case of only substitutions, deletions, or insertions, is received for words that differ on any single bit or variations of the alternating words, here a special behavior can be noticed as the largest intersection is received for words that differ in their middle bits.

In this section, the notation $B_{sub}(\boldsymbol{w})$ for a word $\boldsymbol{w} \in \{0,1\}^*$ will be used to denote the set of words received by exactly a single substitution in $\boldsymbol{w}$. Furthermore, for two words $\boldsymbol{x}, \widetilde{\boldsymbol{x}}$, we say that the diagonal $\searrow$ is maintained on the interval $[k_1, k_2]$, denoted by $\boldsymbol{x}_{[k_1,k_2]} \searrow \widetilde{\boldsymbol{x}}_{[k_1,k_2]}$, if for every bit $k \in [k_1, k_2 - 1]$, $x_k = \widetilde{x}_{k+1}$. Similarly, we say that the diagonal $\nearrow$ is maintained on the interval $[k_1, k_2]$, denoted by $\boldsymbol{x}_{[k_1,k_2]} \nearrow \widetilde{\boldsymbol{x}}_{[k_1,k_2]}$, if for every bit $k \in [k_1+1, k_2]$, $x_k = \widetilde{x}_{k-1}$. If $k_2 = k_1$, then we say that both of the diagonals are maintained in $[k_1, k_2]$.

Before proving the lower and the upper bounds, we find it beneficial to state for which words the intersection of their single-insertion balls is not empty and which words are found in this intersection. We summarize this simple observation in

Lemma 13 and we emphasize that it can be easily extended to discuss intersection of single deletion balls too.

*Lemma 13:* For two words $\boldsymbol{y}, \widetilde{\boldsymbol{y}} \in \{0,1\}^n$, let $j_1, j_2$ be the index of the first, last bit in which $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ differ, respectively. Moreover, let $\boldsymbol{z} \in B_{0,0,1}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$, then one of the following properties holds.

1) $\boldsymbol{z} = y_1 \cdots y_{j_1-1} \widetilde{y}_{j_1} y_{j_1} \cdots y_n$ and the diagonal $\boldsymbol{y} \searrow \widetilde{\boldsymbol{y}}$ is maintained in $[j_1, j_2]$.
2) $\boldsymbol{z} = \widetilde{y}_1 \cdots \widetilde{y}_{j_1-1} y_{j_1} \widetilde{y}_{j_1} \cdots \widetilde{y}_n$ and the diagonal $\boldsymbol{y} \nearrow \widetilde{\boldsymbol{y}}$ is maintained in $[j_1, j_2]$.

*Proof:* First, recall that based on [16] the largest intersection of two single-insertion balls is of size 2. Furthermore, observe that words generated from insertions before $j_1$ in both $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ differ in at least one position, for example $j_1 + 1$. Hence, such pairs of insertions do not generate words in $B_{0,0,1}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$. The same holds for insertions in $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ that are both before $j_2$, both after $j_1$, or both after $j_2$. Therefore, pairs of insertions in $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ that give the same word are insertions that one of them occurred before $j_1$ while the other after $j_2$. Next, consider the following two sets of pairs of insertions in $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$.

1) Insertion in $\boldsymbol{y}$ before $j_1$ and in $\widetilde{\boldsymbol{y}}$ after $j_2$: Let $\boldsymbol{z}$ be a word generated from such insertions. Since the insertion in $\widetilde{\boldsymbol{y}}$ is after $j_2$ (i.e., after $\widetilde{y}_{j_1}$), then $z_{j_1} = \widetilde{y}_{j_1} \neq y_{j_1}$. Thus, an insertion in $\boldsymbol{y}$ that gives $\boldsymbol{z}$ is an insertion of $\widetilde{y}_{j_1}$ exactly before $y_{j_1}$. Similarly, the insertion after $j_2$ in $\widetilde{\boldsymbol{y}}$ should be an insertion of $y_{j_2}$ exactly after $\widetilde{y}_{j_2}$. Hence
$$y_{j_1} = \widetilde{y}_{j_1+1}, y_{j_1+1} = \widetilde{y}_{j_1+2}, \ldots, y_{j_2-1} = \widetilde{y}_{j_2},$$
which means the diagonal $\boldsymbol{y} \searrow \widetilde{\boldsymbol{y}}$ is maintained in section $[j_1, j_2]$.
2) Insertion in $\boldsymbol{y}$ after $j_2$ and in $\widetilde{\boldsymbol{y}}$ before $j_1$: Following the same lines as the previous case, it is possible to conclude that in this set of insertions, there is only one word in $B_{0,0,1}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$. This word can be generated by inserting $y_{j_1}$ exactly before $\widetilde{y}_{j_1}$. Moreover, if this word exists in the intersection then the diagonal $\boldsymbol{y} \nearrow \widetilde{\boldsymbol{y}}$ is maintained in section $[j_1, j_2]$. This proves the second claim in the lemma. ■

The next corollary follows directly from Lemma 13 and determines the condition in which the intersection size of two single-insertion balls is 2.

*Corollary 14:* Given two words $\boldsymbol{y}, \widetilde{\boldsymbol{y}} \in \{0,1\}^n$ such that $|B_{0,0,1}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})| = 2$, let $j_1, j_2$ be the first, last bit in which $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ differ, respectively. Then the sub-words $y_{j_1} \cdots y_{j_2}, \widetilde{y}_{j_1} \cdots \widetilde{y}_{j_2}$ are the alternating words.

Now we are ready to prove the lower bound on $N_n(1,0,1)$. The details of this proof appear in Appendix C.

*Lemma 15:* For $n \geqslant 3$,
$$N_n(1,0,1) \geqslant \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n.$$

Next, we present a decoder in Algorithm 1 that gets $\lfloor \frac{n-2}{2} \rfloor \lceil \frac{n-2}{2} \rceil + 4n + 1$ different words in $B_{1,0,1}(\boldsymbol{w})$ where $\boldsymbol{w} \in \{0,1\}^n, n \geqslant 3$ and recovers $\boldsymbol{w}$ with time complexity $\Theta(n^3)$. Since such an algorithm along with Lemma 15 prove that $N_n(1,0,1) = \lfloor \frac{n-2}{2} \rfloor \lceil \frac{n-2}{2} \rceil + 4n$, which is $\Theta(n^2)$, and since any decoder needs to iterate over all of the words in the worst case, then any decoder will be of time complexity $\Omega(n^3)$. Hence, the decoder's complexity is optimal.

**Algorithm 1** Reconstruct

**Input:** Set $Y \subseteq \{0,1\}^{n+1}$ of size $\lfloor \frac{n-2}{2} \rfloor \lceil \frac{n-2}{2} \rceil + 4n + 1$
**Output:** Word $\boldsymbol{w}$ such that $Y \subseteq B_{1,0,1}(\boldsymbol{w})$

**if** $3 \leqslant n \leqslant 5$ **then**
  **return** the word found by brute-force search
Decode the first, last bit $w_1$, $w_n$ using the majority algorithm on the first, last bit in the erroneous words, respectively
Define the following subsets.
  1) $S_1 = \{\boldsymbol{y} \in Y : y_1 = \overline{w}_1, y_{n+1} = \overline{w}_n\}$
  2) $S_2 = \{\boldsymbol{y} \in Y : y_1 = \overline{w}_1, y_{n+1} = w_n\}$
  3) $S_3 = \{\boldsymbol{y} \in Y : y_1 = w_1, y_{n+1} = \overline{w}_n\}$
  4) $S_4 = \{\boldsymbol{y} \in Y : y_1 = w_1, y_{n+1} = w_n\}$
Let $S_1', S_2', S_3', S_4'$ be the set received from removing the first and the last bit in all of the words of $S_1, S_2, S_3, S_4$, respectively
**if** $|S_1'| = 2$ **then**
  **if** only one word in $S_1'$ begins with $w_1$ **then**
    Let $\boldsymbol{x}$ be the word in $S_1'$ that begins with $\overline{w}_1$
  **else** (both words begin with $w_1$)
    Let $\boldsymbol{x}$ be the word in $S_1'$ with a shorter $w_1$ prefix
  Let $\boldsymbol{z}$ be the prefix of length $n-2$ of $\boldsymbol{x}$
  **return** $w_1 \boldsymbol{z} w_n$
**if** $|S_4'| \geqslant \lfloor \frac{n-4}{2} \rfloor \lceil \frac{n-4}{2} \rceil + 4(n-2) + 1$ **then**
  **return** $w_1$ Reconstruct($S_4'$) $w_n$
Let $M_2 = \emptyset, M_3 = \emptyset$
**if** $|S_2'| > 0$ **then**
  Let $\boldsymbol{x}$ be some word in $S_2'$
  **for** $\boldsymbol{z} \in B_{0,1,0}(\boldsymbol{x}) \cup B_{sub}(x_2\ x_3 \cdots x_{n-1})$ **do**
    **if** $S_2' \subseteq B_{0,0,1}(\boldsymbol{z}) \cup w_1 B_{sub}(\boldsymbol{z})$ **then**
      Add $\boldsymbol{z}$ to $M_2$
**if** $|S_3'| > 0$ **then**
  Let $\boldsymbol{x}$ be some word in $S_3'$
  **for** $\boldsymbol{z} \in B_{0,1,0}(\boldsymbol{x}) \cup B_{sub}(x_1 x_2 \cdots x_{n-2})$ **do**
    **if** $S_3' \subseteq B_{0,0,1}(\boldsymbol{z}) \cup B_{sub}(\boldsymbol{z}) w_n$ **then**
      Add $\boldsymbol{z}$ to $M_3$
**if** $|S_2'| = 0$ **then**
  Let $\boldsymbol{z}$ be the only word in $M_3$
**else if** $|S_3'| = 0$ **then**
  Let $\boldsymbol{z}$ be the only word in $M_2$
**else**
  Let $\boldsymbol{z}$ be the only word in $M_2 \cap M_3$
**return** $w_1 \boldsymbol{z} w_n$

We prove the correctness of Algorithm 1 in the following theorem.

*Theorem 16:* For $n \geqslant 3$, given $\lfloor \frac{n-2}{2} \rfloor \lceil \frac{n-2}{2} \rceil + 4n + 1$ different words in $B_{1,0,1}(\boldsymbol{w})$ where $\boldsymbol{w} \in \{0,1\}^n$, Algorithm 1 returns $\boldsymbol{w}$ with time complexity $\Theta(n^3)$.

*Proof:* As stated in section II, it is sufficient to study the case where the erroneous words result from a single insertion followed by at most a single substitution. In fact, it will be assumed in the rest of this section that the order of the

occurrences is insertion followed by substitution. Next we prove the algorithm's correctness by induction.

Base case: it has been verified by exhaustive search that for binary words of length $n$, where $3 \leqslant n \leqslant 5$, the largest single-insertion single-substitution balls' intersection is of size $\left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n$. Hence, Algorithm 1 returns $\boldsymbol{w}$ correctly in Step 2.

Induction hypothesis: given some $n \geqslant 6$, assume that for any $n'$ where $5 \leqslant n' < n$, Algorithm 1 recovers any $\boldsymbol{w} \in \{0,1\}^{n'}$ using $\left\lfloor \frac{n'-2}{2} \right\rfloor \left\lceil \frac{n'-2}{2} \right\rceil + 4n' + 1$ different words in $B_{1,0,1}(\boldsymbol{w})$ with time complexity $\Theta(n'^3)$.

Induction step: given $\left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 1$ erroneous words in $B_{1,0,1}(\boldsymbol{w})$ where $\boldsymbol{w} \in \{0,1\}^n, n \geqslant 6$, we prove that Algorithm 1 recovers $\boldsymbol{w}$ with time complexity $\Theta(n^3)$.

In Lemma 17, we prove that it is possible to recover the first and the last bit of $\boldsymbol{w}$ by applying the majority algorithm on the corresponding bit in all of the erroneous words, which is an operation of time complexity $\Theta(n^2)$. Moreover, the following statements hold.

1) If $|S_1| \geqslant 2$, then based on Lemma 18, $|S_1| = 2$ and it is possible to recover the word $w_2 \cdots w_{n-1}$ in Steps 7-11 with time complexity $\Theta(n)$.
2) If $|S_4| \geqslant \left\lfloor \frac{n-4}{2} \right\rfloor \left\lceil \frac{n-4}{2} \right\rceil + 4(n-2) + 1$, then according to the induction hypothesis, it is possible to recover the word $w_2 \cdots w_{n-1}$ using Algorithm 1 and the input set $S_4'$ with time complexity $\Theta(n^3)$. It is important to note that since the words in $S_4$ have correct values in the first and the last bits, then the words in $S_4'$ result from a single insertion and at most a single substitution in $w_2 \cdots w_{n-1}$, i.e., $S_4' \subseteq B_{1,0,1}(w_2 \cdots w_{n-1})$.
3) If $|S_4| < \left\lfloor \frac{n-4}{2} \right\rfloor \left\lceil \frac{n-4}{2} \right\rceil + 4(n-2) + 1$ and $|S_1| \neq 2$, then based on Lemma 19, it is possible to recover $w_2 \cdots w_{n-1}$ in Steps 15-31 with time complexity $\Theta(n^3)$.

To sum up, the time complexity to recover the whole word $\boldsymbol{w}$ is upper bounded by

$$\Theta(n^2) + \Theta(n^3) = \Theta(n^3).$$

■

The next lemmas will complete the proof of Theorem 16.

*Lemma 17:* Given at least $\left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 1$ different words in $B_{1,0,1}(\boldsymbol{w})$ where $\boldsymbol{w} \in \{0,1\}^n, n \geqslant 6$, it is possible to decode the first bit $w_1$ using the majority algorithm on the first bit of all of the erroneous words. The same holds for the last bit.

*Proof:* Since errors in the first bit result from either substitution, insertion , or both of them in the first bit, then the set of words in $B_{1,0,1}(\boldsymbol{w})$ that begin with $\overline{w}_1$ is given by the union $\overline{w}_1\ B_{1,0,0}(w_1 \cdots w_n) \cup \overline{w}_1\ B_{0,0,1}(w_2 \cdots w_n)$. Therefore, there are at most $|B_{1,0,0}(w_1 \cdots w_n)| + |B_{0,0,1}(w_2 \cdots w_n)| = (n+1) + (n-1+2) = 2n+2$ words that begin with $\overline{w}_1$. Moreover, for $n \geqslant 6$, it holds that

$$2(2n+2) < \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 1,$$

which means that more than half of the words begin with the correct value $w_1$. Hence, it is possible to recover $w_1$ using the majority algorithm. Regarding the last bit, the claim can be proved similarly. ■

*Lemma 18:* If $|S_1| \geqslant 2$ then $|S_1| = 2$ and it is possible to recover the sub-word $w_2\ w_3 \cdots w_{n-1}$ in Steps 7-11 with time complexity $\Theta(n)$.

*Proof:* Observe that $|S_1| = |S_1'|$, words of $S_1'$ are of length $n+1-2 = n-1$ and can result only by the following ways.

1) Insertion in the first bit and substitution of the last bit of $\boldsymbol{w}$. In this case the erroneous word in $S_1$ is $\overline{w}_1\ w_1\ w_2 \cdots w_{n-1} \overline{w}_n$ and gives the word $w_1\ w_2 \cdots w_{n-1}$ in $S_1'$.
2) Substitution of the first bit and insertion in the last bit of $\boldsymbol{w}$. In this case the erroneous word in $S_1$ is $\overline{w}_1\ w_2 \cdots w_n \overline{w}_n$ and gives the word $w_2 \cdots w_n$ in $S_1'$.

Hence, at least one of the words in $S_1'$ starts with $w_1$ and $|S_1'| \leqslant 2$, which implies that $|S_1'| = 2$. Next consider the following two cases.

1) If there is only one word in $S_1'$ that begins with $w_1$ (Step 8), then it is the word $w_1\ w_2 \cdots w_{n-1}$, so the word $\boldsymbol{x}$ in Step 8 is $\boldsymbol{x} = w_2 \cdots w_n$.
2) If the two words in $S_1'$ begin with $w_1$, then the one with the longer $w_1$ prefix is $w_1\ w_2 \cdots w_{n-1}$ and the word $\boldsymbol{x}$ in Step 10 is $\boldsymbol{x} = w_2 \cdots w_n$.

In both cases, taking the $(n-2)$-prefix of $\boldsymbol{x}$ gives $w_2\ w_3 \cdots w_{n-1}$. Finally, observe that the time complexity of these steps is $\Theta(n)$. ■

*Lemma 19:* If $|S_1| \neq 2$ and $|S_4| < \left\lfloor \frac{n-4}{2} \right\rfloor \left\lceil \frac{n-4}{2} \right\rceil + 4(n-2) + 1$, then it is possible to recover $w_2 \cdots w_{n-1}$ in Steps 15-32 with time complexity $\Theta(n^3)$.

*Proof:* For a word $\boldsymbol{x}$ and a symbol $\gamma$, define the following sets

$$T_{2,\gamma}(\boldsymbol{x}) = B_{0,0,1}(\boldsymbol{x}) \cup \gamma\, B_{sub}(\boldsymbol{x}),$$
$$T_{3,\gamma}(\boldsymbol{x}) = B_{0,0,1}(\boldsymbol{x}) \cup B_{sub}(\boldsymbol{x})\gamma.$$

Moreover, let $M_2', M_3'$ be the set of all of the possibilities for the word $w_2\ w_3 \cdots w_{n-1}$ that can generate the sets $S_2', S_3'$, respectively. If $|S_2'| = \emptyset, |S_3'| = \emptyset$, then let $M_2' = \{0,1\}^{n-2}, M_3' = \{0,1\}^{n-2}$, respectively. Hence, $w_2 w_3 \cdots w_{n-1} \in M_2' \cap M_3'$. Recall that according to the proof of Lemma 18, $|S_1| \leqslant 2$. Hence, if $|S_1| \neq 2$, then $|S_1| < 2$.

Next, it will be proved that under the assumption $|S_1| < 2$ and $|S_4| < \left\lfloor \frac{n-4}{2} \right\rfloor \left\lceil \frac{n-4}{2} \right\rceil + 4(n-2) + 1$, the following properties hold.

1) $|S_2| + |S_3| \geqslant n + 5$.
2) $S_2' \subseteq T_{2,w_1}(w_2 \cdots w_{n-1})$, $S_3' \subseteq T_{3,w_n}(w_2 \cdots w_{n-1})$.
3) $|S_2'|, |S_3'| \leqslant 2n - 2$.
4) If $|S_2'| \geqslant 1, |S_3'| \geqslant 1$, then $M_2 = M_2' = \Theta(n), M_3 = M_3' = \Theta(n)$, respectively. Moreover, $M_2, M_3$ are generated in time complexity $\Theta(n^3)$ in Steps 17-20, 22-25, respectively.
5) If $|S_2'| \geqslant 1$ and $|S_3'| \geqslant 1$, then it is possible to compute the set $M_2 \cap M_3$ in time complexity $\Theta(n^3)$.
6) $|M_2' \cap M_3'| \leqslant 1$.

Here we emphasize that proving these six properties is enough to derive the lemma's statement. According to Property 1, at least one of the sets $S_2, S_3$ is not an empty set. Hence, it is enough to consider the following three cases.

1) $|S_2'| \geqslant 1, |S_3'| = 0$: According to Property 4, at the end of Step 20, $M_2 = M_2'$. Moreover, according to Property 6 and the fact that in this case $M_3' = \{0,1\}^{n-2}$, it holds

that $|M_2'| \leqslant 1$, hence $|M_2| \leqslant 1$. In other words, indeed, in this case only one word will be found in $M_2$ and it is the only possibility for $w_2 w_3 \cdots w_{n-1}$. Hence, in Step 29, $z = w_2 w_3 \cdots w_{n-1}$ and according to Property 4, it is found in time complexity $\Theta(n^3)$.

2) $|S_2'| = 0, |S_3'| \geqslant 1$: As in the case of $|S_2'| \geqslant 1, |S_3'| = 0$, it can be proved that also here, only one word is found in $M_3$ and in Step 27, $z = w_2 w_3 \cdots w_{n-1}$.
3) $|S_2'| \geqslant 1, |S_3'| \geqslant 1$: According to Property 4, at the end of Step 25, $M_2 = M_2'$ and $M_3 = M_3'$. Moreover, according to Property 6, $|M_2 \cap M_3| \leqslant 1$. Hence, there is only one possibility for $w_2 w_3 \cdots w_{n-1}$ found in $M_2 \cap M_3$ and according to Properties 4 and 5, it is generated in time complexity $\Theta(n^3)$.

The six properties stated earlier may be proved as follows.

1) Since $|S_1| < 2$ and $|S_4| < \left\lfloor \frac{n-4}{2} \right\rfloor \left\lceil \frac{n-4}{2} \right\rceil + 4(n-2) + 1$, then it holds that

$$\begin{aligned} |S_2| + |S_3| &\geqslant \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 1 - |S_1| - |S_4| \\ &\geqslant \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 1 \\ &\quad - 1 - \left( \left\lfloor \frac{n-4}{2} \right\rfloor \left\lceil \frac{n-4}{2} \right\rceil + 4(n-2) \right) \\ &= n + 5. \end{aligned}$$

2) Words in $S_2$ have an erroneous value in the first bit and a correct one in the last bit. Therefore, such words may be generated by one of the following ways.
   - Substitution in the first bit and no insertion of $\overline{w}_n$ in the last one. Thus, the insertion occurs in $w_2 \cdots w_{n-1}$.
   - Insertion in the first bit and no substitution in the last one. Thus, the substitution might occur in $w_1 \cdots w_{n-1}$.

   Therefore,

$$S_2' = B_{0,0,1}(w_2 \cdots w_{n-1}) \cup B_{1,0,0}(w_1 w_2 \cdots w_{n-1}).$$

   Since

$$w_1 w_2 \cdots w_{n-1}, \overline{w}_1 w_2 \cdots w_{n-1} \in B_{0,0,1}(w_2 \cdots w_{n-1}),$$

   then the right hand side of the previous equality can be simplified to receive

$$\begin{aligned} S_2' &= B_{0,0,1}(w_2 \cdots w_{n-1}) \cup w_1 B_{sub}(w_2 \cdots w_{n-1}) \\ &= T_{2,w_1}(w_2 \cdots w_{n-1}). \end{aligned}$$

   The proof of $S_3' \subseteq T_{3,w_n}(w_2 \cdots w_{n-1})$ is similar.
3) It holds that

$$\begin{aligned} |w_1 B_{sub}(w_2 \cdots w_{n-1})| &= n-2 \\ |B_{0,0,1}(w_2 \cdots w_{n-1})| &= n-2+2 = n. \end{aligned}$$

   Thus, based on property 2, $|S_2'| = |S_2| \leqslant n-2+n = 2n-2$. A similar proof holds for $|S_3'| \leqslant 2n-2$.
4) Let $\boldsymbol{x} \in S_2'$. It holds that $\boldsymbol{x}$ is generated by one of the $2n-2$ ways specified in $B_{0,0,1}(w_2 \cdots w_{n-1}) \cup w_1 B_{sub}(w_2 \cdots w_{n-1})$. In order to get all of the candidates for $w_2 \cdots w_{n-1}$, i.e., candidates for $M_2'$, we delete any bit of $\boldsymbol{x}$ or substitute a bit of $x_2 \cdots x_{n-1}$ as done in Steps 17-20. For each candidate $z$ we need to check if $S_2' \subseteq B_{0,0,1}(z) \cup w_1\ B_{sub}(z)$, which can be done in time complexity $\Theta(n^2)$. Thus, the total time complexity for Steps 17-20 is

$$(2n-2) \cdot \Theta(n^2) = \Theta(n^3),$$

   and at the end of Step 20, $M_2 = M_2'$. Furthermore, it holds that $|M_2| \leqslant |B_{0,1,0}(\boldsymbol{x}) \cup B_{sub}(x_2 x_3 \cdots x_{n-1})| = \Theta(n)$. The same proof holds for $S_3'$.
5) Given two sets $M_2, M_3$ of size $\Theta(n)$, it is possible to find the set $M_2 \cap M_3$ by going over all of the words in $M_2$. For each word $\boldsymbol{x} \in M_2$, we check if it is found in $M_3$ by going over all of the words in $M_3$ and comparing them with $\boldsymbol{x}$ in $\Theta(n^2)$ time. To sum up, total time complexity is

$$n \cdot \Theta(n^2) = \Theta(n^3).$$

6) It holds that

$$S_2' \subseteq T_{2,w_1}(w_2 \cdots w_{n-1}),\ S_3' \subseteq T_{3,w_n}(w_2 \cdots w_{n-1}).$$

   Since

$$\begin{aligned} &|S_2| + |S_3| = |S_2'| + |S_3'| \\ &\leqslant |T_{2,w_1}(w_2 \cdots w_{n-1})| + |T_{3,w_n}(w_2 \cdots w_{n-1})|, \end{aligned}$$

   and since, based on property 1

$$|S_2| + |S_3| \geqslant n + 5,$$

   then in order to prove that $|M_2' \cap M_3'| = 1$, it is enough to prove that for any two words $\boldsymbol{x}, \widetilde{\boldsymbol{x}} \in \{0,1\}^{n-2}$ and any two symbols $\alpha, \beta \in \{0,1\}$,

$$|T_{2,\alpha}(\boldsymbol{x}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{x}})| + |T_{3,\beta}(\boldsymbol{x}) \cap T_{3,\beta}(\widetilde{\boldsymbol{x}})| < |\boldsymbol{x}| + 7. \tag{2}$$

   Note that

$$\begin{aligned} T_{2,\alpha}(\boldsymbol{x}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{x}}) &= \left(B_{0,0,1}(\boldsymbol{x}) \cap B_{0,0,1}(\widetilde{\boldsymbol{x}})\right) \\ &\quad \cup \left(\alpha B_{sub}(\boldsymbol{x}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{x}})\right) \\ &\quad \cup \left(\alpha B_{sub}(\boldsymbol{x}) \cap B_{0,0,1}(\widetilde{\boldsymbol{x}})\right) \\ &\quad \cup \left(B_{0,0,1}(\boldsymbol{x}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{x}})\right), \\ T_{3,\beta}(\boldsymbol{x}) \cap T_{3,\beta}(\widetilde{\boldsymbol{x}}) &= \left(B_{0,0,1}(\boldsymbol{x}) \cap B_{0,0,1}(\widetilde{\boldsymbol{x}})\right) \\ &\quad \cup \left(B_{sub}(\boldsymbol{x})\beta \cap B_{sub}(\widetilde{\boldsymbol{x}})\beta\right) \\ &\quad \cup \left(B_{sub}(\boldsymbol{x})\beta \cap B_{0,0,1}(\widetilde{\boldsymbol{x}})\right) \\ &\quad \cup \left(B_{0,0,1}(\boldsymbol{x}) \cap B_{sub}(\widetilde{\boldsymbol{x}})\beta\right). \end{aligned}$$

   It is already known by [15] and [16] that

$$|B_{0,0,1}(\boldsymbol{x}) \cap B_{0,0,1}(\widetilde{\boldsymbol{x}})|, |B_{sub}(\boldsymbol{x}) \cap B_{sub}(\widetilde{\boldsymbol{x}})| \leqslant 2.$$

   In Lemma 27 in Appendix D we make several observations regarding the intersections

$$\begin{aligned} &\alpha B_{sub}(\boldsymbol{x}) \cap B_{0,0,1}(\widetilde{\boldsymbol{x}}),\ B_{0,0,1}(\boldsymbol{x}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{x}}), \\ &B_{sub}(\boldsymbol{x})\beta \cap B_{0,0,1}(\widetilde{\boldsymbol{x}}),\ B_{0,0,1}(\boldsymbol{x}) \cap B_{sub}(\widetilde{\boldsymbol{x}})\beta. \end{aligned}$$

   Next, let $\ell = n-2$. In order to prove the inequality in (2), for any two words $\boldsymbol{x}, \widetilde{\boldsymbol{x}} \in \{0,1\}^{\ell}$, let $j_1, j_2$ be the first, last bit for which $x_{j_1} \neq \widetilde{x}_{j_1}, x_{j_2} \neq \widetilde{x}_{j_2}$, respectively. The inequality in (2) has been verified by computer program for all sequences $\boldsymbol{x}, \widetilde{\boldsymbol{x}}$ of length at

most 9. For sequences $\boldsymbol{x}, \widetilde{\boldsymbol{x}}$ of length $\ell \geqslant 10$, we consider the following two cases.

- In $\boldsymbol{x}$, before $j_1$ there are at least two runs or after $j_2$ there are at least two runs. For this case, the inequality in (2) is proved in Lemma 28 in Appendix E.
- In $\boldsymbol{x}$, before $j_1$ there is at most one run, and after $j_2$ there is at most one run. For this case, the inequality in (2) is proved in Lemma 29 in Appendix F. ■

Lastly, the following corollary extends the largest intersection size result to the case of at most one insertion and at most one substitution.

*Corollary 20:* The size of the largest intersection of two balls resulting from at most one insertion and at most one substitution in distinct length-$n$ binary words is given by

$$N_n^*(1,0,1) = N_n(1,0,1) + N_n(1,0,0) = \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 2.$$

The correctness of this corollary is based on the inequality

$$N_n^*(1,0,1) \leqslant N_n(1,0,1) + N_n(1,0,0)$$

and the proof of Lemma 15. For $n \leqslant 2$,

$$B_{1,0,1}^*(0) \cap B_{1,0,1}^*(1) = \{0,1\} \cup \{0,1\}^2,$$

and

$$B_{1,0,1}^*(01) \cap B_{1,0,1}^*(10) = \{00,11\} \cup \{0,1\}^3.$$

Hence, the corollary indeed holds in this case. For $n \geqslant 3$, let

$$\boldsymbol{w} = 0^{\left\lfloor \frac{n-2}{2} \right\rfloor} 100^{\left\lceil \frac{n-2}{2} \right\rceil}$$

and

$$\boldsymbol{w}' = 0^{\left\lfloor \frac{n-2}{2} \right\rfloor} 010^{\left\lceil \frac{n-2}{2} \right\rceil}.$$

Then according to the proof of Lemma 15,

$$\left| B_{1,0,1}(\boldsymbol{w}) \cap B_{1,0,1}(\boldsymbol{w}') \right| = N_n(1,0,1) = \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n.$$

In addition, it holds that

$$\left| B_{1,0,0}(\boldsymbol{w}) \cap B_{1,0,0}(\boldsymbol{w}') \right| = N_n(1,0,0) = 2,$$

which concludes the corollary. Furthermore, given

$$N_n^*(1,0,1) + 1 = \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 3$$

words resulting from at most one insertion and at most one substitution in a length-$n$ binary word $\boldsymbol{w}$, one can recover the word $\boldsymbol{w}$ by the following step.

- If at least 3 length-$n$ words are received, then there are at least 3 words in $B_{1,0,0}(\boldsymbol{w})$. Thus, as proved in [15], $\boldsymbol{w}$ may be recovered by applying the majority algorithm on every bit.
- Otherwise, there are at least

$$\left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 1$$

length-$(n+1)$ binary words. Thus, there are at least

$$\left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n + 1$$

words in $B_{1,0,1}(\boldsymbol{w})$, which allows recovering $\boldsymbol{w}$ using Algorithm 1.

## VI. Decoder for Substitution Errors

In this section, it is studied how to construct a decoder for the reconstruction problem in the substitutions case with optimal complexity. As mentioned in Section II, it was already established by Levenshtein that if the transmitted word belongs to a code with minimum Hamming distance $d$ and there are at most $t$ errors in every channel, then $N_n^S(t,d)+1$ channels are necessary and sufficient to decode in the worst case. However, Levenshtein presented a decoder for this problem only for the case of $d=1$. Since $N_n^S(t,d)+1 = \Theta(n^{t-\lceil \frac{d}{2} \rceil})$, the order of magnitude of the number of bits in any subset $Y \subseteq B_{t,0,0}(x)$ of size $N_n^S(t,d)+1$ is $\Theta(n^{t-\lceil \frac{d}{2} \rceil+1})$. Hence, the complexity order of any decoder is at least this value and a decoder achieving this complexity order will be called *optimal*. A decoder for this problem was presented in [22] for all $d$ and $t$, however its complexity is $\Theta(n^{2t-d})$, and only the case where $d=3$ has been improved in [22] to have an optimal decoder, i.e., with complexity $\Theta(n^{t-1})$. In this section we show how to construct an optimal decoder for all $d$ and $t$. For the rest of this section it is assumed that $d$ and $t$ are fixed positive integers and $n$ is large enough. Furthermore, since for every odd $d$, $N_n^S(t,d) = N_n^S(t,d+1)$ [22], it is also assumed that $d$ is an odd integer and $t > (d-1)/2+1$ (the case $t=(d-1)/2+1$ has been solved in [22]). For shorthand, $N_n^S(t,d)+1$ is denoted by $N_{t,d}$.

Our algorithm uses some of the ideas which were presented in [22] for the decoder in case of $d=3$. A set $Y = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\} \subseteq \{0,1\}^n$ is said to be decoded according to the majority algorithm with threshold $\tau$ such that the algorithm's output is denoted by $\boldsymbol{z} = \mathrm{maj}_\tau(Y)$, if the following condition holds: For all $1 \leqslant i \leqslant n$, let

$$m_{i,0} = |\{j : j \in [N], y_{j,i} = 0\}|, m_{i,1} = |\{j : j \in [N], y_{j,i} = 1\}|,$$

(so $m_{i,0} + m_{i,1} = N$). If $|m_{i,0} - m_{i,1}| \leqslant \tau$ then, $z_i = ?$. Otherwise, if $m_{i,0} > m_{i,1}$ then $z_i = 0$ and if $m_{i,0} < m_{i,1}$ then $z_i = 1$.

For a code $\mathcal{C} \subseteq \{0,1\}^n$ of minimum Hamming distance $d$, we assume that it has a complete decoder $\mathcal{D}_\mathcal{C}$ that can successfully correct at most $\frac{d-1}{2}$ errors. If the number of errors is greater than this value there is no guarantee on the decoder's success. However, a complete decoder outputs a codeword for every input, i.e., even in case there are more than $\frac{d-1}{2}$ errors, the decoder outputs a codeword, yet it is not guaranteed to be the correct one.

In the algorithm, the following value $\tau_{t,d}$ will be used for the threshold of the majority algorithm

$$\tau_{t,d} \triangleq \frac{4}{d+1} \sum_{i=0}^{t-\frac{d+1}{2}} \binom{n-\frac{d+1}{2}}{i} + \frac{d-3}{d+1} N_{t,d}.$$

It is possible to verify that $\tau_{t,d} < N_{t,d}$ for $n$ large enough. We are now ready to present the algorithm. For an integer $t$

and a code $\mathcal{C} \subseteq \{0,1\}^n$ of minimum Hamming distance $d$, we define an algorithm that recovers the transmitted codeword denoted by $\boldsymbol{c}$ using a subset $Y \subseteq B_{t,0,0}(\boldsymbol{c})$ of size $N_{t,d}$.

---

The input to the decoder is a set of all $N_{t,d}$ channels' outputs $Y = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{N_{t,d}}\} \subseteq B_{t,0,0}(\boldsymbol{c})$ for some $\boldsymbol{c} \in \mathcal{C}$ and it returns an estimation $\widehat{\boldsymbol{c}}$ on $\boldsymbol{c}$.

1) $\boldsymbol{z} = \mathrm{maj}_{\tau_{t,d}}(Y)$.
2) $S = \{i : i \in [n], z_i = ?\}$.
3) $Z = \{\boldsymbol{u} \in \{0,1\}^n : u_i = z_i \text{ for all } i \notin S\}$.
4) For all $\boldsymbol{u} \in Z$, $\widehat{\boldsymbol{c}} = \mathcal{D}_{\mathcal{C}}(\boldsymbol{u})$. If $Y \subseteq B_{t,0,0}(\widehat{\boldsymbol{c}})$, output $\widehat{\boldsymbol{c}}$.

---

The correctness of the algorithm is proved using the next two lemmas. For $1 \leqslant i \leqslant n$, let $e_i$ be $e_i = |\{\boldsymbol{y} \in Y : y_i \neq c_i\}|$, i.e., the number of words in $Y$ in which there is an error in the $i$-th position. It holds that, $z_i = ?$ if $\frac{N_{t,d}-\tau_{t,d}}{2} \leqslant e_i \leqslant \frac{N_{t,d}+\tau_{t,d}}{2}$, and $z_i$ is in error if $e_i > \frac{\tau_{t,d}+N_{t,d}}{2}$.

*Lemma 21:* There are at most $\frac{d-1}{2}$ errors in the word $\boldsymbol{z}$ in Step 1.

*Proof:* Assume to the contrary that there exists a set $Y$ such that the word $\boldsymbol{z}$ generated in Step 1 contains at least $\frac{d+1}{2}$ errors. Assume without loss of generality that the first $\frac{d+1}{2}$ bits are erroneous bits. As specified above, for $1 \leqslant i \leqslant \frac{d+1}{2}$

$$e_i \geqslant \frac{N_{t,d}+\tau_{t,d}}{2}+1 = \frac{2}{d+1}\sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}+\frac{d-1}{d+1}N_{t,d}+1.$$

Therefore,

$$\begin{aligned}\sum_{i=1}^{\frac{d+1}{2}} e_i &\geqslant \sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}+\frac{d-1}{2}N_{t,d}+\frac{d+1}{2}\\ &> \sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}+\frac{d-1}{2}N_{t,d}\\ &= \frac{d+1}{2}\sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}+\frac{d-1}{2}\left(N_{t,d}-\sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}\right).\end{aligned}$$

On the other hand, there are at most $\sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}$ words in $Y$ that can have erroneous values in all of the first $\frac{d+1}{2}$ positions. The other $N_{t,d} - \sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}$ words can have at most $\frac{d-1}{2}$ errors in the first $\frac{d+1}{2}$ positions. Therefore, the number of errors in these $\frac{d+1}{2}$ positions is upper bounded by

$$\sum_{i=1}^{\frac{d+1}{2}} e_i \leqslant \frac{d+1}{2}\sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}+\frac{d-1}{2}\left(N_{t,d}-\sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}\right),$$

which results with a contradiction. ∎

*Lemma 22:* For $n$ large enough it holds that $|S| \leqslant t \cdot \left(\frac{d+3}{2}\right)$.

*Proof:* Since in each of the $N_{t,d}$ words in $Y$ there are at most $t$ errors, the total number of errors in all of the $N_{t,d}$ words is bounded by $tN_{t,d}$. Since every erasure requires at least $\frac{N_{t,d}-\tau_{t,d}}{2}$ errors in the copies of this bit, the maximum number of erasures in $\boldsymbol{z}$ is upper bounded by $\frac{tN_{t,d}}{\frac{N_{t,d}-\tau_{t,d}}{2}} = 2t\frac{N_{t,d}}{N_{t,d}-\tau_{t,d}}$. The value $N_{t,d}$ satisfies

$$\begin{aligned}N_{t,d} &= \sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-d}{i}\sum_{h=d-t+i}^{t-i}\binom{d}{h}+1\\ &= \binom{d+1}{\frac{d+1}{2}}\binom{n-d}{t-\frac{d+1}{2}}+\Theta(n^{t-\frac{d+1}{2}-1}),\end{aligned}$$

and for $n$ large enough we have that $N_{t,d} \geqslant \frac{d+3}{2}\sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}$. This implies that

$$\frac{N_{t,d}}{N_{t,d}-\tau_{t,d}} = \frac{N_{t,d}}{\frac{4}{d+1}\left(N_{t,d}-\sum_{i=0}^{t-\frac{d+1}{2}}\binom{n-\frac{d+1}{2}}{i}\right)} \leqslant \frac{d+3}{4},$$

and thus $|S| \leqslant t \cdot \left(\frac{d+3}{2}\right)$. ∎

Lastly, we conclude with the following theorem.

*Theorem 23:* The output $\widehat{\boldsymbol{c}}$ of Algorithm VI is the word $\boldsymbol{c}$. The algorithm's complexity is $\Theta(n^{t-\frac{d+1}{2}+1})$ and hence it is optimal.

*Proof:* From Lemma 21, since there are at most $\frac{d-1}{2}$ errors in the word $\boldsymbol{z}$ of Step 1, one of the $2^{|S|}$ words in the set $Z$ of Step 3 contains at most $\frac{d-1}{2}$ errors. Thus, the decoding of this word in Step 4 will give the correct word $\boldsymbol{c}$. We use here the result of Lemma 16 from [22], in which $\widehat{\boldsymbol{c}} = \boldsymbol{c}$ if and only if $Y \subseteq B_{t,0,0}(\widehat{\boldsymbol{c}})$.

The complexity of Step 1 is $\Theta(N_{t,d}) = \Theta(n^{t-\frac{d+1}{2}+1})$. According to Lemma 22, the size of the sets $S$ and $Z$ is constant with respect to $n$. Thus, the decoding in Step 4 is invoked a constant number of times and the complexity of the condition in this step is $\Theta(n^{t-\frac{d+1}{2}+1})$. Together, we conclude that the algorithm's complexity is $\Theta(n^{t-\frac{d+1}{2}+1})$. We assumed here that the complexity of the decoder $\mathcal{D}_{\mathcal{C}}$ is $O(n^{t-\frac{d+1}{2}+1})$. ∎

## VII. Extensions for the Decoder

Typically, the number of reads for each strand in DNA-based storage systems is different than the value found by Levenshtein. Since this is not a mechanism which is possible to directly control, we study its benefits more specifically, in case there are more reads than the required minimum number of channels, we show how to take advantage of this redundant reads in order to construct a simpler decoder. On the other hand, in case there are less reads than the minimum number, it is not possible to output the correct word in the worst case and then the decoder can only output a list of all potential stored words. While we focus in this work on the first case, the latter has been studied recently in [10].

The simplest decoding algorithm one can think of is the majority decoder in which every bit is decoded by the majority of its copies, that is, $\boldsymbol{z} = \mathrm{maj}_{\tau}(Y)$ for $\tau = 0$. Note that in case the size of $Y$ is even, the algorithm outputs the erasure symbol ? for bits having the same number of one and zero estimations. In fact, this is the decoding algorithm Levenshtein presented for the case of $d = 1$ and any $t$, so the number of channels is $N_n^S(t,1)+1$. This number of channels is necessary for the success of the majority decoder even if the transmitted word belongs to a code of any minimum Hamming distance

$d \geqslant 1$. However, in this case the majority decoder only needs to output a word with at most some $k \leqslant \lfloor \frac{d-1}{2} \rfloor$ errors, as these errors can be corrected by any decoder of the code.

In the following theorems we present bounds on the size of the largest set for which the majority decoder outputs a word with $k+1$ errors.

*Theorem 24:* For $0 < t < k$, $n \in \mathbb{N}$ large enough and a word $\boldsymbol{w} \in \{0,1\}^n$, there exists a set $Y \subseteq B_{t,0,0}(\boldsymbol{w})$ of size

$$M_1 \triangleq 2 \sum_{j=\left\lceil \frac{k+1}{2} \right\rceil}^{t} \binom{k}{j-1} \sum_{i=0}^{t-j} \binom{n-k-1}{i} - 1$$

such that there are at least $k+1$ errors in $\mathrm{maj}_0(Y)$.

*Proof:* Let $Y$ be the set that consists of the following two subsets, $Y_1$ and $Y_2$.

1) The set $Y_1$ consists of all words in $B_{t,0,0}(\boldsymbol{w})$ in which there at least $\left\lceil \frac{k+1}{2} \right\rceil$ errors in the first $k+1$ bits.
2) The set $Y_2$ is of size $M_1 - |Y_1|$ and consists of all words in $B_{t,0,0}(\boldsymbol{w})$ in which there are at most $\left\lceil \frac{k+1}{2} \right\rceil - 1$ errors in the first $k+1$ bits.

In order to prove the theorem, it is enough to show that the set $Y$ is well defined and that the first $k+1$ bits in $\mathrm{maj}_0(Y)$ are erroneous. Hence, it is enough to prove the following claims.

1) $|Y_1| \leqslant M_1$, i.e., the size of the set $Y_2$ is well defined.
2) There are at least $M_1 - |Y_1|$ words in $B_{t,0,0}(\boldsymbol{w})$ in which there are at most $\left\lceil \frac{k+1}{2} \right\rceil - 1$ errors in the first $k+1$ bits, i.e., the set $Y_2$ is well defined.
3) For each bit $i \in [1, k+1]$, there are at least $\left\lceil \frac{|Y|}{2} \right\rceil$ words in $Y$ in which there is an error in the $i$-th bit.

These claims may be proved as follows.

1) The size of the set $Y_1$, as defined in the proof, is

$$\sum_{j=\left\lceil \frac{k+1}{2} \right\rceil}^{t} \binom{k+1}{j} \sum_{i=0}^{t-j} \binom{n-k-1}{i}.$$

Since for $j \geqslant \left\lceil \frac{k+1}{2} \right\rceil$,

$$\binom{k+1}{j} - 2\binom{k}{j-1} = \binom{k}{j} - \binom{k}{j-1} \leqslant 0,$$

then it holds that $|Y_1| \leqslant M_1$.

2) The number of words in $B_{t,0,0}(\boldsymbol{w})$ in which there are at most $\left\lceil \frac{k+1}{2} \right\rceil - 1$ errors in the first $k+1$ bits is equal to

$$M' \triangleq \sum_{j=0}^{\left\lceil \frac{k+1}{2} \right\rceil - 1} \binom{k+1}{j} \sum_{i=0}^{t-j} \binom{n-k-1}{i}.$$

In order to prove that the set $Y_2$ is well defined, it is enough to prove that $M' \geqslant M_1 - |Y_1|$. It holds that $M_1 = \theta(n^{t-\left\lceil \frac{k+1}{2} \right\rceil})$, whereas $M' = \theta(n^{t-\left\lceil \frac{k+1}{2} \right\rceil+1})$. Hence, $M_1 - |Y_1| = O(n^{t-\left\lceil \frac{k+1}{2} \right\rceil})$, and hence for $n$ large enough $M' \geqslant M_1 - |Y_1|$ as required.

3) For all $i \in [1, k+1]$, the number of words in $Y_1$ with an error in the $i$-th position is equal to

$$\sum_{j=\left\lceil \frac{k+1}{2} \right\rceil}^{t} \binom{k}{j-1} \sum_{i=0}^{t-j} \binom{n-k-1}{i},$$

which is exactly $\left\lceil \frac{M_1}{2} \right\rceil$, i.e., $\left\lceil \frac{|Y|}{2} \right\rceil$. Hence, applying the majority algorithm on each of the first $k+1$ bits would result in an erroneous value. Finally, it can be seen that this analysis is correct no matter which words are chosen to be in $Y_2$. ■

While in Theorem 24 we showed a case in which the output of the majority decoder has $k+1$ errors, the next theorem establishes a sufficient condition on the number of channels which guarantees the output to have at most $k$ errors.

*Theorem 25:* For $0 < t < k$, $n \in \mathbb{N}$ large enough, a word $\boldsymbol{w} \in \{0,1\}^n$, and a set $Y \subseteq B_{t,0,0}(\boldsymbol{w})$ of size at least

$$M_2 \triangleq 2 \sum_{j=\left\lceil \frac{k+1}{2} \right\rceil}^{t} \binom{k}{j-1} \sum_{i=0}^{t-j} \binom{n-k-1}{i} + 2k \sum_{j=\left\lceil \frac{k+1}{2} \right\rceil}^{t} \left( \binom{k}{j-1} - \binom{k}{j} \right) \sum_{i=0}^{t-j} \binom{n-k-1}{i} + 1,$$

there are at most $k$ errors in $\mathrm{maj}_0(Y)$.

*Proof:* First, denote by $c_1, c_2$ the value of

$$\sum_{j=\left\lceil \frac{k+1}{2} \right\rceil}^{t} \binom{k}{j-1} \sum_{i=0}^{t-j} \binom{n-k-1}{i},$$

$$\sum_{j=\left\lceil \frac{k+1}{2} \right\rceil}^{t} \left( \binom{k}{j-1} - \binom{k}{j} \right) \sum_{i=0}^{t-j} \binom{n-k-1}{i},$$

respectively. Then, $M_2 = 2c_1 + 2kc_2 + 1$. Assume to the contrary that there exists a set $Z \subseteq B_{t,0,0}(\boldsymbol{w})$ of size $2c_1 + 2kc_2 + m$ where $m \geqslant 1$, for which there are at least $k+1$ errors in $\mathrm{maj}_0(Z)$. Assume without loss of generality that the first $k+1$ values in $\mathrm{maj}_0(Z)$ are erroneous. Hence, for each bit $i \in [1, k+1]$, there are at least $\left\lceil \frac{|Z|}{2} \right\rceil$ words in $Z$ in which there is an erroneous value in the $i$-th position. Therefore, the total number of erroneous values in the first $k+1$ positions in all of the words of $Z$ is lower bounded by

$$M' = (k+1)\left\lceil \frac{|Z|}{2} \right\rceil = (k+1)c_1 + (k+1)kc_2 + (k+1)\left\lceil \frac{m}{2} \right\rceil.$$

Next, an upper bound $M''$ is presented on the total number of erroneous values in the first $k+1$ positions in all of the words of $Z$. It will be shown that $M'' < M'$ which results with a contradiction.

Let $Y_1$ be the set of words in $B_{t,0,0}(\boldsymbol{w})$ that have at least $\left\lceil \frac{k+1}{2} \right\rceil$ errors in the first $k+1$ positions. It holds that

$$|Y_1| = \sum_{j=\left\lceil \frac{k+1}{2} \right\rceil}^{t} \binom{k+1}{j} \sum_{i=0}^{t-j} \binom{n-k-1}{i} = 2c_1 - c_2,$$

and hence $|Y_1| = 2c_1 - c_2 < 2c_1 + 2kc_2 + 1 = M_2$. Moreover, $Y_1$ contributes $c_1$ errors to each of the first $k+1$ positions, thus, $(k+1)c_1$ errors to all of the first $k+1$ positions. Observe

that the number of erroneous values in the first $k+1$ positions in all of the words in $Z$ is upper bounded by

$$M'' \triangleq (k+1)c_1 + \left(\left\lceil \frac{k+1}{2} \right\rceil - 1\right)(|Z| - |Y_1|).$$

Since

$$\begin{aligned} |Z| - |Y_1| &= 2c_1 + 2kc_2 + m - (2c_1 - c_2) \\ &= (2k+1)c_2 + m, \end{aligned}$$

then,

$$\begin{aligned} M'' = (k+1)c_1 &+ \left(\left\lceil \frac{k+1}{2} \right\rceil - 1\right)(2k+1)c_2 \\ &+ \left(\left\lceil \frac{k+1}{2} \right\rceil - 1\right)m. \end{aligned}$$

It is possible to verify that for all $k$ and $m$ it holds that

$$\begin{aligned} \left(\left\lceil \frac{k+1}{2} \right\rceil - 1\right)(2k+1) &< (k+1)k, \\ \left(\left\lceil \frac{k+1}{2} \right\rceil - 1\right)m &< (k+1)\left\lceil \frac{m}{2} \right\rceil \end{aligned}$$

and hence $M'' < M'$, which is a contradiction. ∎

Given $0 < t < k$ and $n \in \mathbb{N}$, let $N_n^{\text{maj}}(t,k)$ be the minimum number of channels such that for all $\boldsymbol{w} \in \{0,1\}^n$ and $Y \subseteq B_{t,0,0}(\boldsymbol{w})$, where $|Y| \geqslant N_n^{\text{maj}}(t,k)$, it holds that $d_H(\boldsymbol{w}, \text{maj}_0(Y)) \leqslant k$. The next corollary summarizes the main result of this section.

*Corollary 26:* For $0 < t < k$, $n \in \mathbb{N}$ large enough it holds that $N_n^{\text{maj}}(t,k) = \Theta(n^{t-\left\lceil \frac{k+1}{2} \right\rceil})$.

As a direct result of Corollary 26, if $d$ is odd, then for $k = \frac{d-1}{2}$, we get that $N_n^{\text{maj}}(t,k) = \Theta(n^{t-\lceil \frac{d+1}{4} \rceil})$, while the minimum number of channels by Levenshtein is $N_n(d,t)+1 = \Theta(n^{t-\frac{d-1}{2}})$. That is, the degree of $n$ in the number of channels increases by $\frac{d-1}{2} - \lceil \frac{d+1}{4} \rceil = \lfloor \frac{d-3}{4} \rfloor$.

## VIII. Conclusion

In this paper we aim to extend the study of the reconstruction problem proposed by Levenshtein to the setup where there are insertions, deletions, and substitutions. In order to initiate this study, we first found the size of the single-deletion $t$-substitution ball and similarly for the single-insertion single-substitution ball. We then continued to find the largest intersection of any two different single-insertion single-substitution balls. Next, we presented an optimal decoder with respect to its complexity for the only substitutions case and lastly the error-correction capability of the majority decoder was studied in case there are more channels than the minimum required number. While significant results are accomplished in this paper, there are still several more open problems under this paradigm. For example, studying the size of the ball under different combinations of insertions, deletions, and substitutions and solving the reconstruction problem stated in Problem 1 for more parameters of $t_1, t_2$, and $t_3$.

## Appendix A

*Claim 1:* The set $T_j$, $j \in [3, 2t]$ consists only of the words that satisfy all of the following requirements.

1) Result from exactly $t$ substitutions in $\boldsymbol{w}^i$.
2) Identical to $\boldsymbol{w}^i$ in positions $i_1, i_2$.
3) Result from at least $\left\lceil \frac{j}{2} \right\rceil$ substitutions in positions $i_1, i_2, \ldots, i_j$ in $\boldsymbol{w}^i$.
4) Result from at most $\lceil \frac{h}{2} \rceil - 1$ substitutions in positions $i_1, i_2, \ldots, i_h$ in $\boldsymbol{w}^i$, for every $3 \leqslant h < j$.

*Proof:* Recall that while considering $T_1, T_2$, words that result from at most $t-1$ substitutions in $\boldsymbol{w}^i$ and words that result from $t$ substitutions in $\boldsymbol{w}^i$ such that at least one of them is in position $i_1$ or $i_2$ are excluded from $B_{t,0,0}(\boldsymbol{w}^i)$. Hence, $T_j$, $j \in [3, 2t]$ consists only of words that result from $t$ substitutions such that none of them is in position $i_1$ or $i_2$. This proves the necessity of the first two requirements.

Recall that $\boldsymbol{w}^i$ differs from $\boldsymbol{w}^{i-j}$ for $j \in [1, 2t]$ in the $j$ positions $i_1, i_2, \ldots, i_j$. Let $\boldsymbol{z}$ be a word that results from applying $t$ substitutions in $\boldsymbol{w}^i$ such that $s$ of them are applied in positions $i_1, i_2, \ldots, i_j$. Then, it holds that

$$d_H(\boldsymbol{w}^{i-j}, \boldsymbol{z}) = j - s + t - s = t + j - 2s$$

as $\boldsymbol{z}$ is identical to $\boldsymbol{w}^{i-j}$ in $s$ positions among $i_1, i_2, \ldots, i_j$. Hence, $\boldsymbol{z}$ differs from $\boldsymbol{w}^{i-j}$ in $j-s$ positions among $i_1, i_2, \ldots, i_j$ and another $t-s$ positions among $[1, n-1] \setminus \{i_1, i_2, \ldots, i_j\}$. Therefore, if $s \geqslant \left\lceil \frac{j}{2} \right\rceil$, then $d_H(\boldsymbol{w}^{i-j}, \boldsymbol{z}) \leqslant t$, i.e., $\boldsymbol{z} \in B_{t,0,0}(\boldsymbol{w}^i) \cap B_{t,0,0}(\boldsymbol{w}^{i-j})$. Otherwise, $d_H(\boldsymbol{w}^{i-j}, \boldsymbol{w}^i) > t$, i.e., $\boldsymbol{z} \notin B_{t,0,0}(\boldsymbol{w}^i) \cap B_{t,0,0}(\boldsymbol{w}^{i-j})$. This proves the necessity of the third requirement.

Since words in $T_j$ are not included in $\bigcup_{h=3}^{j-1} B_{t,0,0}(\boldsymbol{w}^{i-h})$ and since for every $h < j$, $\boldsymbol{w}^i$ differs from $\boldsymbol{w}^{i-h}$ in positions $i_1, i_2, \ldots, i_h$, then based on the same analysis as before, words in $T_j$ are generated by $t$ substitutions in $\boldsymbol{w}^i$ such that for every $h < j$ there are at most $\lceil \frac{h}{2} \rceil - 1$ substitutions in positions $i_1, i_2, \ldots, i_h$. This proves the necessity of the fourth requirement.

To summarize, necessity of all of the four requirements stated in the lemma is now proved. Lastly, we prove that the four requirements are sufficient and we assume that $\boldsymbol{z}$ is a word that satisfies all of the four requirements. Then, based on the first and the second requirements, $\boldsymbol{z} \notin B_{t,0,0}(\boldsymbol{w}^{i-1}) \cup B_{t,0,0}(\boldsymbol{w}^{i-2})$. Moreover, based on the third, fourth requirement, $\boldsymbol{z} \in B_{t,0,0}(\boldsymbol{w}^{i-j})$, $\boldsymbol{z} \notin \bigcup_{h=3}^{j-1} B_{t,0,0}(\boldsymbol{w}^{i-h})$, respectively. Hence, $\boldsymbol{z} \in T_j$. ∎

*Claim 3:* For $j \in [3, 2t]$ even, $|T_j| = C_{\frac{j}{2}-1}\binom{n-1-j}{t-\frac{j}{2}}$.

*Proof:* First we observe that words in $T_j$ satisfy the following two properties.

1) Result from $t$ substitutions in $\boldsymbol{w}^i$ such that exactly $\frac{j}{2}$ are applied in positions $i_1, i_2, \ldots, i_j$, two in positions $i_{j-1}, i_j$ and $\frac{j}{2}-2$ in positions $i_3, i_4, \ldots, i_{j-3}, i_{j-2}$. This property can be proved using the requirements from Claim 1. Words in $T_j$ result from at most $\frac{j-2}{2} - 1 = \frac{j}{2} - 2$ substitutions in $i_1, i_2, \ldots, i_{j-2}$, and hence by at most $\frac{j}{2}$ substitutions in positions $i_1, i_2, \ldots, i_j$. This upper bound, which has to be achieved according to the

third requirement from Claim 1, may be reached only if there are two substitutions in positions $i_{j-1}, i_j$. Moreover, words in $T_j$ are identical to $\boldsymbol{w}^i$ in positions $i_1$ and $i_2$ and hence, $\frac{j}{2}-2$ substitutions are applied in positions $i_3, i_4, \ldots, i_{j-3}, i_{j-2}$.

2) Result from $t$ substitutions in $\boldsymbol{w}^i$ such that for every $h < j$, where $h$ is even, at most $\frac{h}{2}-1$ substitutions are applied in the $h-2$ positions $i_3, i_4, \ldots, i_h$. According to the fourth requirement from Claim 1, it holds that words in $T_j$ result from at most $\lceil \frac{h}{2} \rceil - 1 = \frac{h}{2} - 1$ substitutions in positions $i_1, i_2, \ldots, i_h$. Moreover, there are no substitutions in positions $i_1$ and $i_2$. Thus, there are at most $\frac{h}{2}-1 = \frac{h-2}{2}$ in the $h-2$ positions $i_3, i_4, \ldots, i_h$.

According to these two properties, we conclude that in order to find the size of $T_j$, it is enough to find the number of possibilities for applying $\frac{j-4}{2}$ substitutions in the $j-4$ positions $i_3, i_4, \ldots, i_{j-3}, i_{j-2}$ such that for each $h \in [3, j-2]$ even, there are at most $\frac{h-2}{2}$ substitutions in the $h-2$ positions $i_3, i_4, \ldots, i_h$. If we reinterpret a substitution as a $''$)$''$ and a correct value as $''$($''$, then we see that the number of such possibilities is equal to the number of length-$(j-4)$ words over the alphabet $\{(, )\}$ that have $\frac{j-4}{2}$ $''$($''$'s and $\frac{j-4}{2}$ $''$)$''$'s and satisfy that in each prefix of even length the number of $''$($''$'s is no less than the number of $''$)$''$'s. Denote the set of such words by $W$. Then, there exists a bijective mapping between $W$ and the set of length-$(j-2)$ expressions containing correctly matched $\frac{j}{2}-1$ pairs of parentheses. Hence,

$$|W| = C_{\frac{j-4}{2}+1} = C_{\frac{j}{2}-1}.$$

Finally, note that after finding the number of such possibilities, substitutions in the other positions $[1, n-1] \setminus \{i_1, i_2, \ldots, i_j\}$ should be considered. Thus,

$$|T_j| = C_{\frac{j}{2}-1} \binom{n-1-j}{t-\frac{j}{2}}.$$

■

## APPENDIX B

*Claim 5:* If the $j_1$-th bit falls in the $(h-3)$-rd run or a previous one, then $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) = \emptyset$.

*Proof:* Assume to the contrary that $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) \neq \emptyset$. Then, the $(j_1+2)$-nd bit in $\boldsymbol{w}$ falls in the $(h-1)$-st run or a subsequent one. Otherwise, at least two runs will begin in section $[j_1+2, j_2]$, i.e., according to Claim 4, $d_H(\boldsymbol{w}^{j_1}, \boldsymbol{w}^{j_2}) \geqslant 3$, which contradicts Property 11. Furthermore, we observe that since the $(j_1+2)$-nd position falls in the $(h-1)$-st run or a subsequent one, then the $j_1$-th bit falls in the $(h-3)$-rd run and the $(h-2)$-nd, $(h-1)$-st run begins in the $(j_1+1)$-st, $(j_1+2)$-nd position, respectively. Therefore, according to Claim 4, $d_H(\boldsymbol{w}^{j_1}, \boldsymbol{w}^{j_2}) \geqslant 3$, which contradicts Proposition 11. ■

*Claim 6:* Assume the $j_1$-th bit falls in the $(h-2)$-nd or $(h-1)$-st run. Then, $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) \neq \emptyset$ if and only if the $j_1$-th bit is the last one in its run.

*Proof:* Consider the following two cases.

- The $j_1$-th bit is the last one in its run. In this case $w_{j_1} \neq w_{j_1+1}$. Moreover, at most one run begins in section $[j_1+2, j_2]$. Hence, according to Claim 4, $d_H(\boldsymbol{w}^{j_1}, \boldsymbol{w}^{j_2}) \leqslant 2$, i.e., $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) \neq \emptyset$ according to Proposition 11.
- The $j_1$-th bit is not the last one in its run. In this case $w_{j_1} = w_{j_1+1}$. Moreover, at least one run (for example the $h$-th run) begins in section $[j_1+2, j_2]$. Hence, $d_H(\boldsymbol{w}^{j_1}, \boldsymbol{w}^{j_2}) \geqslant 3$, i.e., $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) = \emptyset$ according to Proposition 11. ■

*Claim 7:* If the $j_1$-th bit falls in the $h$-th run, then $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) \neq \emptyset$.

*Proof:* In this case we observe that $w_{j_1} = w_{j_1+1}$, and that no run begins in section $[j_1+2, j_2]$. Hence, according to Claim 4, $d_H(\boldsymbol{w}^{j_1}, \boldsymbol{w}^{j_2}) = 2$, which means that $B_{1,0,0}(\boldsymbol{w}^{j_1}) \cap B_{1,0,0}(\boldsymbol{w}^{j_2}) \neq \emptyset$ according to Proposition 11. ■

*Claim 8:* For $i \geqslant 1$, the following statements hold.

- $B_{1,0,0}(\boldsymbol{w}^{-1}) \cap B_{1,0,0}(\boldsymbol{w}^i) \neq \emptyset$ if and only if the $i$-th bit falls in the first or the second run.
- $B_{1,0,0}(\boldsymbol{w}^0) \cap B_{1,0,0}(\boldsymbol{w}^i) \neq \emptyset$ if and only if the $i$-th bit falls in the first run.

*Proof:* The two statements may be proved as follows.

- Let the $i$-th bit in $\boldsymbol{w}$ fall in the third run or a subsequent one. Then according to Claim 4, $\boldsymbol{w}^{-1}$ differs from $\boldsymbol{w}^i$ in positions $\ell_1+1, \ell_1+\ell_2+1$ and $i+1$. Hence, $d_H(\boldsymbol{w}^{-1}, \boldsymbol{w}^i) \geqslant 3$, and thus $B_{1,0,0}(\boldsymbol{w}^{-1}) \cap B_{1,0,0}(\boldsymbol{w}^i) = \emptyset$ according to Proposition 11. Furthermore, if the $i$-th bit falls in the first, second run, then $\boldsymbol{w}^{-1}$ can be received by inserting $w_i, \overline{w_i}$ at the beginning of the first, second run, respectively, i.e., $d_H(\boldsymbol{w}^{-1}, \boldsymbol{w}^i) \leqslant 2$. Hence, $B_{1,0,0}(\boldsymbol{w}^{-1}) \cap B_{1,0,0}(\boldsymbol{w}^i) \neq \emptyset$ according to Proposition 11.
- Let the $i$-th bit in $\boldsymbol{w}$ fall in the second run or a subsequent one. Then according to Claim 4, $\boldsymbol{w}^0$ differs from $\boldsymbol{w}^i$ in positions $1, \ell_1+1$ and $i+1$. Hence, $d_H(\boldsymbol{w}^0, \boldsymbol{w}^i) \geqslant 3$, so $B_{1,0,0}(\boldsymbol{w}^0) \cap B_{1,0,0}(\boldsymbol{w}^i) = \emptyset$ according to Proposition 11. Furthermore, if the $i$-th bit falls in the first run, then both of $\boldsymbol{w}^i$ and $\boldsymbol{w}^0$ are received by insertions to the same run. Hence, $d_H(\boldsymbol{w}^0, \boldsymbol{w}^i) \leqslant 2$ and $B_{1,0,0}(\boldsymbol{w}^0) \cap B_{1,0,0}(\boldsymbol{w}^i) \neq \emptyset$. ■

*Claim 9:* Assume $B_{1,0,0}(\boldsymbol{w}^{i_1}) \cap B_{1,0,0}(\boldsymbol{w}^{i_2}) \neq \emptyset$ where $-1 \leqslant i_1 < i_2 \leqslant n$ and the $i_2$-th bit in $\boldsymbol{w}$ falls in the $h'$-th run, then $\boldsymbol{w}^{i_1}$ can be received by some insertion to the $h'$-th run, including insertion of $w_{i_2}$ or $\overline{w}_{i_2}$ at the beginning of the $h'$-th run.

*Proof:* Assume $B_{1,0,0}(\boldsymbol{w}^{i_1}) \cap B_{1,0,0}(\boldsymbol{w}^{i_2}) \neq \emptyset$. If $i_1 > 0$, then according to Claims 5 and 6, $i_1$ falls in either the $(h'-2)$-nd, $(h'-1)$-st, or the $h'$-th run. In case it falls in the $h'$-th run then the proof is immediate. In case it falls in the $(h'-2)$-nd, $(h'-1)$-st run, then $i_1$ is the last bit in its run, and hence $\boldsymbol{w}^{i_1}$ can be received by inserting $\overline{w}_{i_2}, w_{i_2}$ at the beginning of the $h'$-th, respectively.

If $i_1 \leqslant 0$, then according to Claim 8, $\boldsymbol{w}^{i_1}$ can be received by some insertion to the $h'$-th run. ■

## APPENDIX C

*Lemma 15.:* For $n \geqslant 3$,

$$N_n(1,0,1) \geqslant \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n.$$

*Proof:* It has been verified by a computer program that for $n = 3$, $N_n(1,0,1) = \lfloor \frac{n-2}{2} \rfloor \lceil \frac{n-2}{2} \rceil + 4n$. Henceforth, assume

$n \geqslant 4$. Let $\boldsymbol{y}, \boldsymbol{y}'$ be the word $0^{\left\lfloor \frac{n-2}{2} \right\rfloor}, 0^{\left\lceil \frac{n-2}{2} \right\rceil}$, respectively. Moreover, let $\boldsymbol{w}, \boldsymbol{w}'$ be the word $\boldsymbol{y}10\boldsymbol{y}'$, $\boldsymbol{y}01\boldsymbol{y}'$, respectively. We prove that

$$|B_{1,0,1}(\boldsymbol{w}) \cap B_{1,0,1}(\boldsymbol{w}')| = \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n.$$

Define the following sets:

$$\begin{aligned}
S'_{000} &= B_{1,0,0}(\boldsymbol{y})\, 000\boldsymbol{y}', \\
S''_{000} &= \boldsymbol{y}000\ B_{1,0,0}(\boldsymbol{y}'), \\
S_{001} &= \{\boldsymbol{y}001\boldsymbol{y}'\}, \\
S_{010} &= B_{1,0,0}(\boldsymbol{y})\, 010\ B_{1,0,0}(\boldsymbol{y}'), \\
S_{011} &= B_{1,0,0}(\boldsymbol{y})\, 011\boldsymbol{y}', \\
S_{100} &= \{\boldsymbol{y}100\boldsymbol{y}'\}, \\
S'_{101} &= B_{1,0,0}(\boldsymbol{y})\, 101\boldsymbol{y}', \\
S''_{101} &= \boldsymbol{y}101\ B_{1,0,0}(\boldsymbol{y}'), \\
S_{110} &= \boldsymbol{y}110\ B_{1,0,0}(\boldsymbol{y}'), \\
S_{111} &= \{\boldsymbol{y}111\boldsymbol{y}'\}.
\end{aligned}$$

Observe that the intersection $S'_{000} \cap S''_{000}$, $S'_{101} \cap S''_{101}$ consists of the word $\boldsymbol{y}000\boldsymbol{y}'$, $\boldsymbol{y}101\boldsymbol{y}'$, respectively. Furthermore, note that the sets $S'_{000} \cup S''_{000}$, $S_{001}$, $S_{010}$, $S_{011}$, $S_{100}$, $S'_{101} \cup S''_{101}$, $S_{110}$ and $S_{111}$ differ in positions $\left\lfloor \frac{n-2}{2} \right\rfloor + 1$, $\left\lfloor \frac{n-2}{2} \right\rfloor + 2$, $\left\lfloor \frac{n-2}{2} \right\rfloor + 3$, and hence are disjoint sets and of the following sizes.

$$\begin{aligned}
&|S'_{000} \cup S''_{000}| = n - 2 + 1, \\
&|S_{001}| = |S_{100}| = |S_{111}| = 1, \\
&|S_{010}| = 1 + n - 2 + \left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil, \\
&|S_{011}| = \left\lfloor \frac{n-2}{2} \right\rfloor + 1, \\
&|S'_{101} \cup S''_{101}| = n - 2 + 1, \\
&|S_{110}| = \left\lceil \frac{n-2}{2} \right\rceil + 1,
\end{aligned}$$

Hence, the union of these sets is of size

$$\left\lfloor \frac{n-2}{2} \right\rfloor \left\lceil \frac{n-2}{2} \right\rceil + 4n,$$

i.e., it is enough to prove that $B_{1,0,1}(\boldsymbol{w}) \cap B_{1,0,1}(\boldsymbol{w}')$ is equal to the union of these sets.

Next, observe that

$$B_{1,0,1}(\boldsymbol{w}) \cap B_{1,0,1}(\boldsymbol{w}') = \bigcup_{\substack{z \in B_{1,0,0}(\boldsymbol{w}) \\ z' \in B_{1,0,0}(\boldsymbol{w}')}} B_{0,0,1}(z) \cap B_{0,0,1}(z').$$

Since based on Lemma 13, for every two words $z, z' \in \{0,1\}^n$, $B_{0,0,1}(z) \cap B_{0,0,1}(z') = \emptyset$ if and only if no diagonal is maintained between the the first and the last positions in which $z, z'$ differ, then it is enough to consider substitutions that maintain at least one diagonal.

Let $z, z'$ be a word generated from at most one substitution in $\boldsymbol{w}, \boldsymbol{w}'$, respectively. Moreover, if $d_H(z, \boldsymbol{w}) = 1, d_H(z', \boldsymbol{w}') = 1$, then assume the $m_1, m_2$-th bit is the substituted one in $z, z'$, respectively. Consider the following cases.

1) $d_H(z, \boldsymbol{w}) = 0, d_H(z', \boldsymbol{w}') = 0$. In this case,
$$\begin{aligned} B_{0,0,1}(z) \cap B_{0,0,1}(z') &= \{\boldsymbol{y}010\boldsymbol{y}', \boldsymbol{y}101\boldsymbol{y}'\} \\ &\subseteq S_{010} \cup S'_{101} \cup S''_{101}. \end{aligned}$$

2) $d_H(z, \boldsymbol{w}) = 1, d_H(z', \boldsymbol{w}') = 0$. In this case, one of the diagonals is maintained if and only if one of the following properties holds.
- $m_1 = \left\lfloor \frac{n-2}{2} \right\rfloor + 1$, in this case,
$$\begin{aligned} B_{0,0,1}(z) \cap B_{0,0,1}(z') &= \{\boldsymbol{y}010\boldsymbol{y}', \boldsymbol{y}001\boldsymbol{y}'\} \\ &\subseteq S_{010} \cup S_{001}. \end{aligned}$$
- $m_1 = \left\lfloor \frac{n-2}{2} \right\rfloor + 2$-th position, in this case,
$$\begin{aligned} B_{0,0,1}(z) \cap B_{0,0,1}(z') &= \{\boldsymbol{y}011\boldsymbol{y}', \boldsymbol{y}101\boldsymbol{y}'\} \\ &\subseteq S'_{101} \cup S''_{101} \cup S_{011}. \end{aligned}$$
- $m_1 > \left\lfloor \frac{n-2}{2} \right\rfloor + 2$, in this case,
$$\begin{aligned} B_{0,0,1}(z) \cap B_{0,0,1}(z') &= \boldsymbol{y}010\ B_{sub}(\boldsymbol{y}') \cup \{\boldsymbol{y}101\boldsymbol{y}'\} \\ &\subseteq S_{010} \cup S'_{101} \cup S''_{101}. \end{aligned}$$

3) $d_H(z, \boldsymbol{w}) = 0, d_H(z', \boldsymbol{w}') = 1$. In this case, one of the diagonals is maintained if and only if one of the following properties holds.
- $m_2 = \left\lfloor \frac{n-2}{2} \right\rfloor + 1$, in this case,
$$\begin{aligned} B_{0,0,1}(z) \cap B_{0,0,1}(z') &= \{\boldsymbol{y}101\boldsymbol{y}', \boldsymbol{y}110\boldsymbol{y}'\} \\ &\subseteq S_{110} \cup S'_{101} \cup S''_{101}. \end{aligned}$$
- $m_2 = \left\lfloor \frac{n-2}{2} \right\rfloor + 2$, in this case,
$$\begin{aligned} B_{0,0,1}(z) \cap B_{0,0,1}(z') &= \{\boldsymbol{y}010\boldsymbol{y}', \boldsymbol{y}100\boldsymbol{y}'\} \\ &\subseteq S_{010} \cup S_{100}. \end{aligned}$$
- $m_2 < \left\lfloor \frac{n-2}{2} \right\rfloor + 1$, in this case,
$$\begin{aligned} B_{0,0,1}(z) \cap B_{0,0,1}(z') &= B_{sub}(\boldsymbol{y})\, 010\boldsymbol{y}' \cup \{\boldsymbol{y}101\boldsymbol{y}'\} \\ &\subseteq S_{010} \cup S'_{101} \cup S''_{101}. \end{aligned}$$

4) $d_H(z, \boldsymbol{w}) = 1, d_H(z', \boldsymbol{w}') = 1$ and $m_1, m_2 \in \left[\left\lfloor \frac{n-2}{2} \right\rfloor + 1, \left\lfloor \frac{n-2}{2} \right\rfloor + 2\right]$. In this case, one of the diagonals is maintained if and only if one of the following properties holds.
   a) $m_1 = \left\lfloor \frac{n-2}{2} \right\rfloor + 1, m_2 = \left\lfloor \frac{n-2}{2} \right\rfloor + 2$, in this case,
$$\begin{aligned} B_{0,0,1}(z) \cap B_{0,0,1}(z') &= S'_{000} \cup S''_{000} \cup S_{100} \cup S_{001} \\ &\quad \cup \{\boldsymbol{y}010\boldsymbol{y}'\} \subseteq S'_{000} \cup S''_{000} \\ &\quad \cup S_{100} \cup S_{001} \cup S_{010}. \end{aligned}$$
   b) $m_1 = \left\lfloor \frac{n-2}{2} \right\rfloor + 2, m_2 = \left\lfloor \frac{n-2}{2} \right\rfloor + 1$, in this case,
$$\begin{aligned} B_{0,0,1}(z) \cap B_{0,0,1}(z') &= S_{011} \cup S_{110} \cup S_{111} \\ &\quad \cup \{\boldsymbol{y}101\boldsymbol{y}'\} \subseteq S'_{101} \cup S''_{101} \\ &\quad \cup S_{011} \cup S_{110} \cup S_{111}. \end{aligned}$$

5) $d_H(z, \boldsymbol{w}) = 1, d_H(z', \boldsymbol{w}') = 1$ and one of $m_1, m_2$ is in $\left[\left\lfloor \frac{n-2}{2} \right\rfloor + 1, \left\lfloor \frac{n-2}{2} \right\rfloor + 2\right]$ while the other is not. In this case, one of the diagonals is maintained if and only if one of the following properties holds.

a) $m_2 = \lfloor \frac{n-2}{2} \rfloor + 1, m_1 = \lfloor \frac{n-2}{2} \rfloor$, in this case,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = \{y110y'\} \subseteq S_{110}.$$

b) $m_2 = \lfloor \frac{n-2}{2} \rfloor + 1, m_1 > \lfloor \frac{n-2}{2} \rfloor + 2$, in this case,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = y110\ B_{sub}(y') \cup \{y101y'\} \subseteq S_{110} \cup S'_{101} \cup S''_{101}.$$

c) $m_1 = \lfloor \frac{n-2}{2} \rfloor + 2, m_2 = \lfloor \frac{n-2}{2} \rfloor + 3$, in this case,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = \{y011y'\} \subseteq S_{011}.$$

d) $m_1 = \lfloor \frac{n-2}{2} \rfloor + 2, m_2 < \lfloor \frac{n-2}{2} \rfloor + 1$, in this case,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = B_{sub}(y)\,011y' \cup \{y101y'\} \subseteq S_{011} \cup S'_{101} \cup S''_{101}.$$

6) $d_H(z, w) = 1, d_H(z', w') = 1$ and $m_1, m_2 < \lfloor \frac{n-2}{2} \rfloor + 1$. In this case, one of the diagonals is maintained if and only if one of the following properties holds.

a) $m_1 = m_2$, in this case,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = B_{sub}(y)\,101y' \cup B_{sub}(y)\,010y' \subseteq S'_{101} \cup S_{010}.$$

b) $m_2 = \lfloor \frac{n-2}{2} \rfloor$, in this case,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = \left\{0^{\lfloor \frac{n-2}{2} \rfloor - 1}1010y'\right\} \cup B_{sub}(y)\,101y' \subseteq S_{010} \cup S'_{101}.$$

c) $m_1 + 1 = m_2$, in this case,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = 0B_{sub}\left(0^{\lfloor \frac{n-2}{2} \rfloor - 1}\right)010y' \subseteq S_{010}.$$

7) $d_H(z, w) = 1, d_H(z', w') = 1$ and $m_1, m_2 > \lfloor \frac{n-2}{2} \rfloor + 2$. In this case, one of the diagonals is maintained if and only if one of the following properties holds.

a) $m_1 = m_2$, in this case,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = y101\ B_{sub}(y') \cup y010\ B_{sub}(y') \subseteq S''_{101} \cup S_{010}.$$

b) $m_1 = \lfloor \frac{n-2}{2} \rfloor + 3$, in this case,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = \left\{y01010^{\lceil \frac{n-2}{2} \rceil - 1}\right\} \cup y101\ B_{sub}(y') \subseteq S_{010} \cup S''_{101}.$$

c) $m_1 + 1 = m_2$, in this case,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = y010\ B_{sub}\left(0^{\lceil \frac{n-2}{2} \rceil - 1}\right)0 \subseteq S_{010}.$$

8) $d_H(z, w) = 1, d_H(z', w') = 1$ and one of $m_1, m_2$ is less than $\lfloor \frac{n-2}{2} \rfloor + 1$ while the other is greater than $\lfloor \frac{n-2}{2} \rfloor + 2$. In this case, one of the diagonals is maintained only if $m_1 > \lfloor \frac{n-2}{2} \rfloor + 2$ and $m_2 < \lfloor \frac{n-2}{2} \rfloor + 1$, hence,

$$B_{0,0,1}(z) \cap B_{0,0,1}(z') = B_{sub}(y)\,010\ B_{sub}(y') \cup \{y101y'\} \subseteq S_{010} \cup S'_{101} \cup S''_{101}.$$

Considering all of the cases above, it is possible to verify that all of the words in the union $S'_{000} \cup S''_{000} \cup S_{001} \cup S_{010} \cup S_{011} \cup S_{100} \cup S'_{101} \cup S''_{101} \cup S_{110} \cup S_{111}$ belong to the intersection $B_{1,0,1}(w) \cap B_{1,0,1}(w')$. ■

## APPENDIX D

*Lemma 27:* Let $y, \widetilde{y} \in \{0,1\}^\ell$ where $\ell \geqslant 4$. Let $j_1, j_2$ be the first, last index in which $y$ and $\widetilde{y}$ differ, respectively. Furthermore, let $r_1$ be the length of the last run in $y_1 \cdots y_{j_1-1}$ (if $j_1 = 1$ then $r_1 = 0$) and similarly, $r_2$ is the length of the first run in $y_{j_2+1} \cdots y_\ell$ (if $j_2 = \ell$ then $r_2 = 0$). Define $i_1, i_2$ to be the following values

$$i_1 = j_1 - 1 - r_1,$$
$$i_2 = j_2 + r_2 + 1.$$

The following upper bounds hold

$$|B_{0,0,1}(y) \cap \alpha B_{sub}(\widetilde{y})| \leqslant \begin{cases} 1, & y_1 = \overline{\alpha} \\ 2, & y_1 = \alpha \text{ and the diagonal } y \nearrow \widetilde{y} \text{ is not maintained in } [1, j_2] \\ i_2 - j_2, & y_1 = \alpha \text{ and the diagonal } y \nearrow \widetilde{y} \text{ is maintained in } [1, i_2 - 1] \\ 1, & y_1 = \alpha, \text{ the diagonal } y \nearrow \widetilde{y} \text{ is maintained in } [1, j_2] \text{ and broken in } [j_2, j_2+1] \end{cases}$$

$$|B_{0,0,1}(y) \cap B_{sub}(\widetilde{y})\beta| \leqslant \begin{cases} 1, & y_n = \overline{\beta} \\ 2, & y_n = \beta \text{ and the diagonal } y \searrow \widetilde{y} \text{ is not maintained in } [j_1, \ell] \\ j_1 - i_1, & y_n = \beta \text{ and the diagonal } y \searrow \widetilde{y} \text{ is maintained in } [i_1 + 1, \ell] \\ 1, & y_n = \beta, \text{ the diagonal } y \searrow \widetilde{y} \text{ is maintained in } [j_1, \ell] \text{ and broken in } [j_1 - 1, j_1] \end{cases}$$

*Proof:* We prove only the upper bound on $|B_{0,0,1}(y) \cap \alpha B_{sub}(\widetilde{y})|$, while the proof for $|B_{0,0,1}(y) \cap B_{sub}(\widetilde{y})\beta|$ follows the same lines. First, it is possible to verify that the cases given in the lemma are disjoint and include all of the possible pairs $y, \widetilde{y}$.

Consider the following cases.

1) $y_1 = \overline{\alpha}$. In this case there is only one word in $B_{0,0,1}(y)$ that begins with $\alpha$. Thus, $|B_{0,0,1}(y) \cap \alpha B_{sub}(\widetilde{y})| \leqslant 1$. Note that such a word exists only if $d_H(y, \widetilde{y}) = 1$.
2) $y_1 = \alpha$ and the diagonal $y \nearrow \widetilde{y}$ is not maintained in $[1, j_2]$, i.e., for some bit $1 < k \leqslant j_2$, $y_k \neq \widetilde{y}_{k-1}$.

Let $k'$ be the first bit for which this inequality holds. Consider the following two subsets that together compose $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$.

- $R_1 = \{\boldsymbol{z} : \boldsymbol{z} \in B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}}), z_{k'} = y_{k'}\}$. Observe that in order to receive such words, the $(k'-1)$-st bit in $\widetilde{\boldsymbol{y}}$ should be substituted. Thus, $|R_1| \leqslant |\{\boldsymbol{z} : \boldsymbol{z} \in \alpha B_{sub}(\widetilde{\boldsymbol{y}}), z_{k'} = y_{k'}\}| = 1$.
- $R_2 = \{\boldsymbol{z} : \boldsymbol{z} \in B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}}), z_{k'} = \overline{y}_{k'} = \widetilde{y}_{k'-1}\}$. Observe that in order to receive such a word an insertion in $\boldsymbol{y}$ should be applied before the $k'$-th bit. Let $\boldsymbol{z}$ be some word in $R_2$, i.e., received by some insertion before the $k'$-th bit in $\boldsymbol{y}$ so it holds that $z_{j_2+1} = y_{j_2}$. On the other hand, the $(j_2+1)$-st bit in $\alpha\widetilde{\boldsymbol{y}}$ is equal to $\widetilde{y}_{j_2}$. Hence, $\boldsymbol{z}$ and $\alpha\widetilde{\boldsymbol{y}}$ differ in the $(j_2+1)$-st bit, which means that in order to get $\boldsymbol{z}$ from $\widetilde{\boldsymbol{y}}$, a specific substitution should be applied which is the substitution in the $j_2$-th bit. Thus, $R_2 \subseteq \{\alpha\widetilde{y}_1 \cdots \widetilde{y}_{j_2-1} y_{j_2} \widetilde{y}_{j_2+1} \cdots \widetilde{y}_\ell\}$. i.e., $|R_2| \leqslant 1$.

Since $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}}) = R_1 \cup R_2$, then $|B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})| \leqslant 2$ as requested.

3) $y_1 = \alpha$ and the diagonal $\boldsymbol{y} \nearrow \widetilde{\boldsymbol{y}}$ is maintained in $[1, i_2 - 1]$. Note that in this case, based on the definition of the diagonal $\boldsymbol{y} \nearrow \widetilde{\boldsymbol{y}}$, $y_{j_2+1} = \widetilde{y}_{j_2+1} = \widetilde{y}_{j_2}$ and $y_2 \cdots y_{j_2} = \widetilde{y}_1 \cdots \widetilde{y}_{j_2-1}$. Since $y_1 = \alpha$, then $y_1 \cdots y_{j_2} = \alpha\widetilde{y}_1 \cdots \widetilde{y}_{j_2-1}$. Next, we will show that $|B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})| = i_2 - j_2$ by proving that words in $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ can be received by substituting exactly one of the bits in positions $[j_2, i_2 - 1]$ in $\widetilde{\boldsymbol{y}}$ and no other bit in $\widetilde{\boldsymbol{y}}$. Hence, we will prove that words in $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ differ from $\alpha\widetilde{\boldsymbol{y}}$ in the section $[j_2+1, i_2]$ and are identical to $\alpha\widetilde{\boldsymbol{y}}$ in sections $[1, j_2]$ and $[i_2+1, \ell+1]$. We can prove that by the following three claims.

- There exist $i_2 - j_2$ words in $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$. This claim can be proved by observing that all of the $i_2 - j_2$ words generated by exactly one substitution in section $[j_2+1, i_2]$ in $\alpha\widetilde{\boldsymbol{y}}$ can be also received by specific insertions in positions $[j_2, i_2-1]$ in $\boldsymbol{y}$, i.e., these $i_2 - j_2$ words are in $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$.
- All of the words in $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ have the same length-$j_2$ prefix as the word $\alpha\widetilde{\boldsymbol{y}}$, i.e., the same length-$j_2$ prefix as the word $\boldsymbol{y}$. If $j_2 = 1$, then the claim holds as all of the words in $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ begin with $\alpha$. Henceforth, assume that $j_2 > 1$. In this case, the claim can be proved by assuming to the contrary that there is a word in $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ with a different length-$j_2$ prefix. Such a word can be received only by insertions before the $j_2$-th bit in $\boldsymbol{y}$ and substitution of one of the bits in section $[1, j_2-1]$ in $\widetilde{\boldsymbol{y}}$. Let $\boldsymbol{z}$ be such a word. Then, the $(j_2+1)$-st bit in $\boldsymbol{z}, \alpha\widetilde{\boldsymbol{y}}$ is equal to $y_{j_2}, \widetilde{y}_{j_2}$, respectively, which means that $\boldsymbol{z}$ and $\alpha\widetilde{\boldsymbol{y}}$ differ in the $(j_2+1)$-st bit. Therefore, $d_H(\boldsymbol{z}, \alpha\widetilde{\boldsymbol{y}}) \geqslant 2$, which contradicts the definition of $B_{sub}$.
- If $i_2 = \ell+1$, then the words of $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ are of length $i_2$. Otherwise, all of the words in $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ have the same suffix that begins in the $(i_2+1)$-st bit as the suffix of $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ that begin in the $i_2$-th bit. This claim can be proved by assuming to the contrary that there exists a word in $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ that has a different suffix. Such a word can be received only by substitution in bits $[i_2, \ell]$ in $\widetilde{\boldsymbol{y}}$ and insertion after the $i_2$-th bit in $\boldsymbol{y}$. Let $\boldsymbol{z}$ be such a word. It holds that the $i_2$-th bit in $\boldsymbol{z}, \alpha\widetilde{\boldsymbol{y}}$ is equal to $y_{i_2}, \widetilde{y}_{i_2-1}$. Since after the $j_2$-th bit, $\boldsymbol{y}$ and $\widetilde{\boldsymbol{y}}$ are identical, and since $i_2$ is a position in which a new run begins, then $\boldsymbol{z}$ and $\alpha\widetilde{\boldsymbol{y}}$ differ in the $i_2$-th bit. Thus, $d_H(\boldsymbol{z}, \alpha\widetilde{\boldsymbol{y}}) \geqslant 2$, which contradicts the definition of $B_{sub}$.

4) $y_1 = \alpha$, the diagonal $\boldsymbol{y} \nearrow \widetilde{\boldsymbol{y}}$ is maintained in $[1, j_2]$ and is broken in $[j_2, j_2+1]$. First, note that based on the definition of the diagonal $\boldsymbol{y} \nearrow \widetilde{\boldsymbol{y}}$, $y_{j_2+1} \neq \widetilde{y}_{j_2}$, i.e., $y_{j_2+1} = \widetilde{y}_{j_2+1} = y_{j_2}$. Moreover, as stated in the previous case, $y_1 \cdots y_{j_2} = \alpha\widetilde{y}_1 \cdots \widetilde{y}_{j_2-1}$ and $y_{j_2+1} \cdots y_\ell = \widetilde{y}_{j_2+1} \cdots \widetilde{y}_\ell$. Consider the following two subsets that together compose $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$.

- $R_1 = \{\boldsymbol{z} :\in B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}}), z_{j_2+1} = \widetilde{y}_{j_2}\}$. Observe that words in $R_1$ can be received only by insertion of $\widetilde{y}_{j_2}$ exactly before $y_{j_2+1}$ in $\boldsymbol{y}$. However, such an insertion would give the word $\alpha\widetilde{\boldsymbol{y}}$, which contradicts the definition of $B_{sub}$.
- $R_2 = \{\boldsymbol{z} :\in B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}}), z_{j_2+1} = y_{j_2}\}$. Observe that in order to get words in $R_2$, a specific substitution should be applied in $\widetilde{\boldsymbol{y}}$ which is substitution of the $j_2$-th. Hence, $R_2 \subseteq \{\alpha\widetilde{y}_1 \cdots \widetilde{y}_{j_2-1} y_{j_2} \widetilde{y}_{j_2+1} \cdots \widetilde{y}_\ell\}$, i.e., $|R_2| \leqslant 1$.

Since $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}}) = R_1 \cup R_2$, then $|B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})| \leqslant 1$. ■

## APPENDIX E

*Lemma 28:* Given two binary words $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ of length $\ell \geqslant 10$, let $j_1, j_2, i_1, i_2$ be as defined in Lemma 27. Assume that $i_1 > 0$ or $i_2 < \ell + 1$ (i.e., either before $j_1$ there are two runs or after $j_2$ there are two runs). It holds that for every $\alpha, \beta \in \{0, 1\}$, $|T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})| + |T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})| < \ell + 7$.

*Proof:* We prove the lemma by induction.

Base case: The claim has been verified by exhaustive search for words of length at most 9 as mentioned at the end of the proof of Lemma 19.

Induction hypothesis: Let $\ell \geqslant 10$. Assume that for all words $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ of length $\ell' < \ell$ for which $i_1 > 0$ or $i_2 < \ell' + 1$, the following inequality holds

$$|T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})| + |T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})| < \ell' + 7.$$

Inductive step: Let $\ell \geqslant 10$. We prove that for length-$\ell$ words $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ for which $i_1 > 0$ or $i_2 < \ell + 1$, it holds that

$$|T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})| + |T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})| < \ell + 7.$$

Assume to the contrary that there are $\alpha, \beta$ and two words $\boldsymbol{y}, \widetilde{\boldsymbol{y}} \in \{0, 1\}^\ell$ for which $i_1 > 0$ or $i_2 < \ell + 1$, however,

$$|T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})| + |T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})| \geqslant \ell + 7.$$

Assume without loss of generality that $i_1 > 0$. Then, there are at least two runs before $j_1$, i.e., $j_1 > 1$ and $y_1 = \widetilde{y}_1$. We show that in this case

$$|T_{2,y_1}(y_2 \cdots y_\ell) \cap T_{2,y_1}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)| + |T_{3,\beta}(y_2 \cdots y_\ell) \cap T_{3,\beta}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)| \geqslant \ell + 7.$$

Note that $y_2 \cdots y_\ell$ and $\widetilde{y}_2 \cdots \widetilde{y}_\ell$ are words of length $\ell - 1$. If $y_2 \cdots y_\ell$ and $\widetilde{y}_2 \cdots \widetilde{y}_\ell$ consist of at most one run before the first different bit and at most one run after the last different bit, then this conclusion contradicts Lemma 29. Otherwise, it contradicts the induction hypothesis.

Let $T_2', T_3'$ be the set received by removing the first bit in all of the words in $T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})$, $T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})$, respectively. In the rest of the proof, we prove the correctness of the following four claims.

1) $T_2' \subseteq T_{2,y_1}(y_2 \cdots y_\ell) \cap T_{2,y_1}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)$.
2) $T_3' \subseteq T_{3,\beta}(y_2 \cdots y_\ell) \cap T_{3,\beta}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)$.
3) $|T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})| = |T_2'|$.
4) $|T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})| = |T_3'|$.

By these four claims, it is possible to conclude that

$$\begin{aligned}\ell - 1 + 8 = \ell + 7 &\leqslant |T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})| \\ &\quad + |T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})| = |T_2'| + |T_3'| \\ &\leqslant |T_{2,y_1}(y_2 \cdots y_\ell) \cap T_{2,y_1}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)| \\ &\quad + |T_{3,\beta}(y_2 \cdots y_\ell) \cap T_{3,\beta}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)|,\end{aligned}$$

which results with a contradiction. We proceed with proving the last four claims.

1) We show that for all $\boldsymbol{z} \in T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})$, it holds that $z_2 \cdots z_{\ell+1} \in T_{2,y_1}(y_2 \cdots y_\ell) \cap T_{2,y_1}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)$. Recall that the intersection $T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})$ consists of the following four subsets.
   - $B_{0,0,1}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$: Observe that if $\boldsymbol{z} \in B_{0,0,1}(\boldsymbol{y})$ then $z_2 \cdots z_{\ell+1} \in B_{0,0,1}(y_2 \cdots y_\ell)$, and the same holds for $\widetilde{\boldsymbol{y}}$. Hence, if $\boldsymbol{z} \in B_{0,0,1}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$, then $z_2 \cdots z_{\ell+1} \in B_{0,0,1}(y_2 \cdots y_\ell) \cap B_{0,0,1}(\widetilde{y}_2 \cdots \widetilde{y}_\ell))$.
   - $\alpha B_{sub}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$: Words in this intersection are received only from substituting the bits that differ between $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$. Since $y_1 = \widetilde{y}_1$ then such words are identical to $\alpha \boldsymbol{y}$ and $\alpha \widetilde{\boldsymbol{y}}$ in the first and the second bit and are different from $\alpha \boldsymbol{y}, \alpha \widetilde{\boldsymbol{y}}$ in positions in $[3, \ell+1]$. Hence, if $\boldsymbol{z} \in \alpha B_{sub}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ then $z_2 \cdots z_{\ell+1} \in y_1\ B_{sub}(y_2 \cdots y_\ell) \cap y_1\ B_{sub}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)$.
   - $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$: Assume that $\boldsymbol{z}$ belongs to this intersection, then $z_2 \cdots z_{\ell+1} \in B_{0,0,1}(y_2 \cdots y_\ell)$. Moreover, either $z_2 \cdots z_{\ell+1} = \overline{y}_1 \widetilde{y}_2 \cdots \widetilde{y}_\ell \in B_{0,0,1}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)$ or $z_2 \cdots z_{\ell+1} \in y_1\ B_{sub}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)$. Thus,

$$\begin{aligned}z_2 \cdots z_{\ell+1} \in &\left(B_{0,0,1}(y_2 \cdots y_\ell) \cap B_{0,0,1}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)\right) \\ &\cup \left(B_{0,0,1}(y_2 \cdots y_\ell) \cap y_1\ B_{sub}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)\right).\end{aligned}$$

   - $\alpha B_{sub}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$: If $\boldsymbol{z}$ belongs to this intersection, then

$$\begin{aligned}z_2 \cdots z_{\ell+1} \in &\left(B_{0,0,1}(y_2 \cdots y_\ell) \cap B_{0,0,1}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)\right) \\ &\cup \left(y_1\ B_{sub}(y_2 \cdots y_\ell) \cap B_{0,0,1}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)\right).\end{aligned}$$

     for the same explanation as in the previous case.
2) As in the proof of the first claim, in this proof we consider the four subsets that comprise $T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})$. In case $\boldsymbol{z}$ belongs to the intersection $B_{0,0,1}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$ or the intersection $B_{sub}(\boldsymbol{y})\beta \cap B_{sub}(\widetilde{\boldsymbol{y}})\beta$, then it is possible to prove that $z_2 \cdots z_{\ell+1} \in T_{3,\beta}(y_2 \cdots y_\ell) \cap T_{3,\beta}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)$ in the same way as in the previous claim. Next, assume that $\boldsymbol{z} \in B_{0,0,1}(\boldsymbol{y}) \cap B_{sub}(\widetilde{\boldsymbol{y}})\beta$. In this case, $z_2 \cdots z_{\ell+1} \in B_{0,0,1}(y_2 \cdots y_\ell)$. Moreover, according to Lemma 27, $\boldsymbol{z}$ differs from $\widetilde{\boldsymbol{y}}\beta$ only in bits in the run preceding $j_1$. Since there are two runs before $j_1$ and $y_1$, $\widetilde{y}_1$ are not in the run that comes exactly before $j_1$, then they are not substituted, i.e., $\boldsymbol{z}$ differs from $\widetilde{\boldsymbol{y}}\beta$ in positions in the interval $[2, \ell]$. Thus, $z_2 \cdots z_{\ell+1} \in B_{sub}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)\beta$, which means that $z_2 \cdots z_{\ell+1} \in B_{0,0,1}(y_2 \cdots y_\ell) \cap B_{sub}(\widetilde{y}_2 \cdots \widetilde{y}_\ell)\beta$. A similar proof holds in case $\boldsymbol{z} \in B_{sub}(\boldsymbol{y})\beta \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$.
3) Here it is proved that there are no two words in $T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})$ that differ only in the first bit. Assume to the contrary that there are two words $\boldsymbol{x}, \widetilde{\boldsymbol{x}} \in T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})$ that differ only in the first bit (thus removing the first bit results in the same word $x_2 \cdots x_\ell$). Since $x_1 \neq \widetilde{x}_1$, then one of them is $\alpha$ while the other is $\overline{\alpha}$. Assume without loss of generality that $x_1 = \alpha$ and $\widetilde{x}_1 = \overline{\alpha}$. According to the definition of $T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})$, we have that

$$\boldsymbol{x}, \widetilde{\boldsymbol{x}} \in T_{2,\alpha}(\boldsymbol{y}) = B_{0,0,1}(\boldsymbol{y}) \cup \alpha B_{sub}(\boldsymbol{y}).$$

   Since $\widetilde{x}_1 = \overline{\alpha}$, then $\widetilde{\boldsymbol{x}} \notin \alpha B_{sub}(\boldsymbol{y})$, i.e., $\widetilde{\boldsymbol{x}} \in B_{0,0,1}(\boldsymbol{y})$. Furthermore, we have the following observations.
   a) If $\boldsymbol{x} \in B_{0,0,1}(\boldsymbol{y})$ then both $\boldsymbol{x}, \widetilde{\boldsymbol{x}}$ were received by a single insertion to $\boldsymbol{y}$. Since $\boldsymbol{x}$ and $\widetilde{\boldsymbol{x}}$ differ only in the first bit and both are generated from insertions to the same word then, the insertions were in the first position, i.e., $\boldsymbol{x} = \alpha \boldsymbol{y}, \widetilde{\boldsymbol{x}} = \overline{\alpha} \boldsymbol{y}$. To sum up, in this case $\boldsymbol{y} = x_2 \cdots x_{\ell+1}$. We will refer to this conclusion by (*).
   b) If $\boldsymbol{x} \in \alpha B_{sub}(\boldsymbol{y})$ then

$$x_2 \cdots x_{\ell+1} \in B_{sub}(\boldsymbol{y}), \widetilde{\boldsymbol{x}} \in B_{0,0,1}(\boldsymbol{y}).$$

      Consider the following two cases.
      - $y_1 = \widetilde{y}_1 = \alpha$, then

$$\boldsymbol{y} = \widetilde{x}_2 \cdots \widetilde{x}_{\ell+1} = x_2 \cdots x_{\ell+1},$$

        in contradiction with $x_2 \cdots x_{\ell+1} \in B_{sub}(\boldsymbol{y})$. In summary, if $\boldsymbol{x} \in \alpha B_{sub}(\boldsymbol{y})$, then $y_1 = \widetilde{y}_1 = \overline{\alpha}$. We denote this conclusion by (**).
      - $y_1 = \widetilde{y}_1 = \overline{\alpha}$, then

$$\begin{aligned}x_2 \cdots x_{\ell+1} = \widetilde{x}_2 \cdots \widetilde{x}_{\ell+1} \in B_{0,0,1}(y_2 \cdots y_\ell), \\ \widetilde{x}_2 \cdots \widetilde{x}_{\ell+1} = x_2 \cdots x_{\ell+1} \in B_{sub}(\boldsymbol{y}).\end{aligned}$$

        Thus, the substitution in $\boldsymbol{x}$ is in the first run of $\boldsymbol{y}$ (because only substitutions in the first run in $\boldsymbol{y}$ can be achieved by single insertion to $y_2 \cdots y_\ell$). We denote this conclusion by (***).

   Now based on the conclusions above, consider the following cases.
   a) $\boldsymbol{x}, \widetilde{\boldsymbol{x}} \in B_{0,0,1}(\boldsymbol{y}), \boldsymbol{x}, \widetilde{\boldsymbol{x}} \in B_{0,0,1}(\widetilde{\boldsymbol{y}})$, then based on (*)

$$\boldsymbol{y} = x_2 \cdots x_{\ell+1} = \widetilde{x}_2 \cdots \widetilde{x}_{\ell+1} = \widetilde{\boldsymbol{y}}.$$

   b) $\boldsymbol{x}, \widetilde{\boldsymbol{x}} \in B_{0,0,1}(\boldsymbol{y}), \boldsymbol{x} \in \alpha B_{sub}(\widetilde{\boldsymbol{y}}), \widetilde{\boldsymbol{x}} \in B_{0,0,1}(\widetilde{\boldsymbol{y}})$. According to (**), $y_1 = \widetilde{y}_1 = \overline{\alpha}$. Then

$$\boldsymbol{y} = x_2 \cdots x_{\ell+1} = \widetilde{x}_2 \cdots \widetilde{x}_{\ell+1}.$$

      According to (***), the word $x_2 \cdots x_{\ell+1}$ is generated from substitution in the first run in $\widetilde{\boldsymbol{y}}$. Thus, $\boldsymbol{y}$ differs

from $\widetilde{\boldsymbol{y}}$ by one bit in the first run, which is a contradiction to the fact that $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ are identical in the first run.

c) $\boldsymbol{x}, \widetilde{\boldsymbol{x}} \in B_{0,0,1}(\widetilde{\boldsymbol{y}}), \boldsymbol{x} \in \alpha B_{sub}(\boldsymbol{y}), \widetilde{\boldsymbol{x}} \in B_{0,0,1}(\boldsymbol{y})$. The proof follows similar analysis as the case b.

d) $\boldsymbol{x} \in \alpha B_{sub}(\boldsymbol{y}), \widetilde{\boldsymbol{x}} \in B_{0,0,1}(\boldsymbol{y}), \boldsymbol{x} \in \alpha B_{sub}(\widetilde{\boldsymbol{y}}), \widetilde{\boldsymbol{x}} \in B_{0,0,1}(\widetilde{\boldsymbol{y}})$. According to (**), $y_1 = \widetilde{y}_1 = \overline{\alpha}$. Moreover, according to (***), the word $x_2 \cdots x_{\ell+1}$ is received from substitutions in the first run in both $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$. However, the first run in $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ is identical, so $x_2 \cdots x_{\ell+1}$ is generated from the same substitution in the first run of both $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$. This implies that the rest of $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ is identical too. Thus, we receive a contradiction.

4) Next, it is proved that all the words in $T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})$ begin with $y_1 = \widetilde{y}_1$. Recall that this intersection consists of the following four subsets, and for each subset it is proved that all of its words begin with $y_1 = \widetilde{y}_1$.
   - $B_{0,0,1}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$: Assume to the contrary that there is a word $\boldsymbol{z} \in B_{0,0,1}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$ beginning with $\overline{y}_1$, then such a word is generated by inserting $\overline{y}_1$ at the beginning of $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$, i.e., $\boldsymbol{z} = \overline{y}_1 \boldsymbol{y} = \overline{y}_1 \widetilde{\boldsymbol{y}}$. However, $d_H(\overline{y}_1 \boldsymbol{y}, \overline{y}_1 \widetilde{\boldsymbol{y}}) = d_H(\boldsymbol{y}, \widetilde{\boldsymbol{y}}) > 0$.
   - $B_{sub}(\boldsymbol{y})\beta \cap B_{sub}(\widetilde{\boldsymbol{y}})\beta$: Substitutions in this intersection occur only in the bits that differ between $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$. Thus, there is no substitution in the first bit.
   - $B_{0,0,1}(\boldsymbol{y}) \cap B_{sub}(\widetilde{\boldsymbol{y}})\beta$: According to Lemma 27, substitutions will be only in the run preceding $j_1$, thus not in the first bit. Hence, all of these words begin with $y_1 = \widetilde{y}_1$.
   - $B_{sub}(\boldsymbol{y})\beta \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$: Same explanation as for the case $B_{0,0,1}(\boldsymbol{y}) \cap B_{sub}(\widetilde{\boldsymbol{y}})\beta$. ■

## APPENDIX F

In the following lemma, for any two words $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ and any two symbols $\alpha, \beta \in \{0, 1\}$, we use the following shortcuts:

| | |
|---|---|
| $B_{0,0,1}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$ | $I, I$ |
| $B_{sub}(\boldsymbol{y}) \cap B_{sub}(\widetilde{\boldsymbol{y}})$ | $S, S$ |
| $\alpha B_{sub}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ | $\alpha S, \alpha S$ |
| $B_{sub}(\boldsymbol{y})\beta \cap B_{sub}(\widetilde{\boldsymbol{y}})\beta$ | $S\beta, S\beta$ |
| $\alpha B_{sub}(\boldsymbol{y}) \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$ | $\alpha S, I$ |
| $B_{0,0,1}(\boldsymbol{y}) \cap \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ | $I, \alpha S$ |
| $B_{sub}(\boldsymbol{y})\beta \cap B_{0,0,1}(\widetilde{\boldsymbol{y}})$ | $S\beta, I$ |
| $B_{0,0,1}(\boldsymbol{y}) \cap B_{sub}(\widetilde{\boldsymbol{y}})\beta$ | $I, S\beta$ |
| $\lvert T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})\rvert + \lvert T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})\rvert$ | $Sum$ |

*Lemma 29:* Given two words $\boldsymbol{y}, \widetilde{\boldsymbol{y}} \in \{0, 1\}^\ell$ where $\ell \geqslant 10$, let $j_1, j_2$ be the first, last bit for which the two words $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ differ. Assume that before $j_1$ there is at most one run in $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$, and after $j_2$ there is at most one run in $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$, then for any two symbols $\alpha, \beta \in \{0, 1\}$, the following holds

$$\lvert T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})\rvert + \lvert T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})\rvert < \ell + 7.$$

*Proof:* First, it holds that

$$T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}}) = (I, I) \cup (\alpha S, \alpha S) \cup (\alpha S, I) \cup (I, \alpha S),$$
$$T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}}) = (I, I) \cup (S\beta, S\beta) \cup (S\beta, I) \cup (I, S\beta).$$

Hence,

$$Sum \leqslant 2|I, I| + 2|S, S| + |\alpha S, I| + |I, \alpha S| + |S\beta, I| + |I, S\beta|.$$

Second, we make the following observations which will be useful in the rest of the proof. Their proofs are omitted since they repeat previous ideas.

*Proposition 30:* If $j_1 > 1$, then at most one diagonal is maintained in the section $[1, j_1]$. Similarly, if $j_2 < \ell$, then at most one diagonal is maintained in the section $[j_2, \ell]$.

*Proposition 31:* Both of the diagonals are maintained in the sections $[1, j_1 - 1]$ and $[j_2 + 1, \ell]$.

*Proposition 32:* If $j_1 > 1$, $j_2 < \ell$, $y_1 = \widetilde{y}_1 = \overline{\alpha}$ and $y_\ell = \widetilde{y}_\ell = \overline{\beta}$, then $Sum \leqslant 12$.

*Proof:* According to Lemma 27, the following upper bounds hold

| $\lvert I, I\rvert$ | $\lvert S, S\rvert$ | $\lvert I, \alpha S\rvert$ | $\lvert \alpha S, I\rvert$ | $\lvert I, S\beta\rvert$ | $\lvert S\beta, I\rvert$ | $Sum$ |
|---|---|---|---|---|---|---|
| 2 | 2 | 1 | 1 | 1 | 1 | 12 |

■

Next let $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ be two words as stated in the lemma, and let $c_1, c_2$ be the number of diagonals maintained in the section $[1, j_2]$, $[j_1, \ell]$, respectively. For $\ell \leqslant 9$, the lemma is proved by exhaustive search as mentioned at the end of the proof of Lemma 19. For $\ell > 9$, we consider all of the following cases and for each one we prove that $Sum < \ell + 7$.

1) $c_1 = 0, c_2 = 0$. In this case, based on Lemma 27, the following upper bounds hold

| $\lvert I, I\rvert$ | $\lvert S, S\rvert$ | $\lvert I, \alpha S\rvert$ | $\lvert \alpha S, I\rvert$ | $\lvert I, S\beta\rvert$ | $\lvert S\beta, I\rvert$ | $Sum$ |
|---|---|---|---|---|---|---|
| 2 | 2 | 2 | 2 | 2 | 2 | 16 |

2) $c_1 = 1, c_2 = 0$. Assume without loss of generality that the only diagonal maintained in $[1, j_2]$ is $\boldsymbol{y} \searrow \widetilde{\boldsymbol{y}}$, then $\boldsymbol{y} \searrow \widetilde{\boldsymbol{y}}$ is maintained also in $[j_1, j_2]$, however, it is broken in $[j_2, \ell]$. Hence, according to Proposition 31, the diagonal $\boldsymbol{y} \searrow \widetilde{\boldsymbol{y}}$ is broken in $[j_2, j_2 + 1]$. To sum up, the diagonal $\boldsymbol{y} \searrow \widetilde{\boldsymbol{y}}$ is the only one maintained in $[1, j_2]$, it is broken in $[j_2, j_2 + 1]$, and no diagonal is maintained in $[j_1, \ell]$. Therefore, according to Lemma 27, the following upper bounds hold.

| $\lvert I, I\rvert$ | $\lvert S, S\rvert$ | $\lvert I, \alpha S\rvert$ | $\lvert \alpha S, I\rvert$ | $\lvert I, S\beta\rvert$ | $\lvert S\beta, I\rvert$ | $Sum$ |
|---|---|---|---|---|---|---|
| 2 | 2 | 2 | 1 | 2 | 2 | 15 |

3) $c_1 = 0, c_2 = 1$. The proof is similar to the case of $c_1 = 1, c_2 = 0$.

4) $c_1 = 2, c_2 = 0$. Observe that only one diagonal can be broken in $[j_2, j_2 + 1]$. Hence, a diagonal is broken in $[j_2 + 1, \ell]$, which contradicts Proposition 31. Therefore, $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$, cannot satisfy $c_1 = 2, c_2 = 0$ and hence there is no need to consider this case.

5) $c_1 = 0, c_2 = 2$. The proof is similar to the case of $c_1 = 2, c_2 = 0$.

6) $c_1 \geqslant 1, c_2 \geqslant 1$. Consider the following cases.

   a) $j_1 = 1, j_2 = \ell$. In this case, according to Lemma 27, the following upper bounds hold

| $\lvert I, I\rvert$ | $\lvert S, S\rvert$ | $\lvert I, \alpha S\rvert$ | $\lvert \alpha S, I\rvert$ | $\lvert I, S\beta\rvert$ | $\lvert S\beta, I\rvert$ | $Sum$ |
|---|---|---|---|---|---|---|
| 2 | 2 | 2 | 2 | 2 | 2 | 16 |

   b) $j_1 = 1, j_2 < \ell$. Assume without loss of generality that $y_1 = \overline{\alpha}, \widetilde{y}_1 = \alpha$. The following properties hold.
      - If the diagonal $\boldsymbol{y} \searrow \widetilde{\boldsymbol{y}}$ is not maintained in $[j_1, j_2]$, then the following upper bounds hold

| $\lvert I, I\rvert$ | $\lvert S, S\rvert$ | $\lvert I, \alpha S\rvert$ | $\lvert \alpha S, I\rvert$ | $\lvert I, S\beta\rvert + \lvert S\beta, I\rvert$ | $Sum$ |
|---|---|---|---|---|---|
| 2 | 2 | 1 | 2 | 4 | 15 |

- Exactly one diagonal is maintained in $[j_1, \ell]$. This claim can be proved as follows. According to Proposition 30, at most one diagonal is maintained in $[j_2, \ell]$, i.e., $c_2 \leqslant 1$. Moreover, $c_2 \geqslant 1$. Hence, $c_2 = 1$.
- If the diagonal $\boldsymbol{y} \searrow \widetilde{\boldsymbol{y}}$ is maintained in $[j_1, j_2]$ and not in $[j_1, \ell]$, then according to Proposition 31, it is broken in $[j_2 + 1, j_2]$. In this case, according to Lemma 27, the following upper bounds hold

| $\lvert I, I\rvert$ | $\lvert S, S\rvert$ | $\lvert I, \alpha S\rvert$ | $\lvert \alpha S, I\rvert$ | $\lvert I, S\beta\rvert + \lvert S\beta, I\rvert$ | $Sum$ |
|---|---|---|---|---|---|
| 2 | 2 | 1 | 1 | 4 | 14 |

Considering all of these claims, it is enough to deal with words $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ in which the diagonal $\boldsymbol{y} \searrow \widetilde{\boldsymbol{y}}$ is maintained in all of the word, while the diagonal $\boldsymbol{y} \nearrow \widetilde{\boldsymbol{y}}$ may be maintained only in $[1, j_2]$ and may not. Note that since the diagonal $\boldsymbol{y} \searrow \widetilde{\boldsymbol{y}}$ is maintained in $[j_2, j_2 + 1]$, and since $y_{j_2} \neq \widetilde{y}_{j_2}$ and $y_{j_2+1} = \widetilde{y}_{j_2+1}$, then the diagonal $\boldsymbol{y} \nearrow \widetilde{\boldsymbol{y}}$ is not maintained in $[j_2, j_2 + 1]$. Let $k = j_2 - j_1 + 1$, consider the following cases.

- $k = 1$. In this case $\boldsymbol{y} = \overline{\alpha}(\overline{\alpha})^h, \widetilde{\boldsymbol{y}} = \alpha(\overline{\alpha})^h$ where $h \geqslant 1$. Let

$$\begin{aligned} z_1 &= \alpha\overline{\alpha}(\overline{\alpha})^h, \\ z_2 &= \overline{\alpha}\alpha(\overline{\alpha})^h, \\ z_3 &= \alpha\alpha(\overline{\alpha})^h, \\ z_4 &= \overline{\alpha}(\overline{\alpha})^h\overline{\alpha}, \\ z_5 &= \alpha(\overline{\alpha})^h\alpha, \\ z_6 &= \overline{\alpha}(\overline{\alpha})^h\alpha. \end{aligned}$$

Then,

$$\begin{aligned} &I, I = \{z_1, z_2\}, \\ &\alpha S, \alpha S = \{z_1, z_3\}, \\ &I, \alpha S = \{z_1\}, \\ &\alpha S, I = \alpha B_{sub}\left(\overline{\alpha}(\overline{\alpha})^h\right). \end{aligned}$$

If $\beta = \overline{\alpha}$, then

$$\begin{aligned} &S\beta, S\beta = \{z_4, z_1\}, \\ &S\beta, I = \{z_1, z_2\}, \\ &I, S\beta = \{z_4\}. \end{aligned}$$

Otherwise,

$$\begin{aligned} &S\beta, S\beta = \{z_5, z_6\}, \\ &S\beta, I = \{z_5\}, \\ &I, S\beta = \{z_6\}. \end{aligned}$$

Since, $z_3 \in \alpha S, I$ and since

$$|\alpha S, I| = \left|B_{sub}\left(\overline{\alpha}(\overline{\alpha})^h\right)\right| = \ell,$$

then

$$\begin{aligned} \left|T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})\right| &= |I, I \cup \alpha S, \alpha S \cup \alpha S, I \cup I, \alpha S| \\ &= \ell + 2, \\ \left|T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})\right| &= |I, I \cup S\beta, S\beta \cup S\beta, I \cup I, S\beta| \\ &\leqslant 4, \end{aligned}$$

which means that $Sum \leqslant \ell + 6$.

- $k = 2$. In this case $\boldsymbol{y} = \overline{\alpha}\alpha\alpha^h, \widetilde{\boldsymbol{y}} = \alpha\overline{\alpha}\alpha^h$ where $h \geqslant 1$. Let

$$\begin{aligned} z_1 &= \alpha\overline{\alpha}\alpha\alpha^h, \\ z_2 &= \overline{\alpha}\alpha\overline{\alpha}\alpha^h, \\ z_3 &= \alpha\overline{\alpha\alpha}\alpha^h, \\ z_4 &= \alpha\alpha\alpha\alpha^h, \\ z_5 &= \overline{\alpha\alpha}\alpha^h\alpha, \\ z_6 &= \overline{\alpha\alpha}\alpha^h\overline{\alpha}, \\ z_7 &= \alpha\alpha\alpha^h\overline{\alpha}. \end{aligned}$$

Then,

$$\begin{aligned} &I, I = \{z_1, z_2\}, \\ &\alpha S, \alpha S = \{z_3, z_4\}, \\ &I, \alpha S = \emptyset, \\ &\alpha S, I = \alpha\overline{\alpha}B_{sub}\left(\alpha\alpha^h\right). \end{aligned}$$

If $\beta = \alpha$, then

$$\begin{aligned} &S\beta, S\beta = \{z_4, z_5\}, \\ &S\beta, I = \{z_2\}, \\ &I, S\beta = \{z_5\}. \end{aligned}$$

Otherwise,

$$\begin{aligned} &S\beta, S\beta = \{z_6, z_7\}, \\ &S\beta, I = I, S\beta = \emptyset. \end{aligned}$$

Since $z_3 \in \alpha S, I$ and since

$$|\alpha S, I| = \left|B_{sub}\left(\alpha\alpha^h\right)\right| = \ell - 1,$$

then

$$\begin{aligned} \left|T_{2,\alpha}(\boldsymbol{y}) \cap T_{2,\alpha}(\widetilde{\boldsymbol{y}})\right| &= |I, I \cup \alpha S, \alpha S \cup \alpha S, I \cup I, \alpha S| \\ &= \ell + 2, \\ \left|T_{3,\beta}(\boldsymbol{y}) \cap T_{3,\beta}(\widetilde{\boldsymbol{y}})\right| &= |I, I \cup S\beta, S\beta \cup S\beta, I \cup I, S\beta| \\ &= 4, \end{aligned}$$

which means $Sum = \ell + 6$.

- $k \geqslant 3$. In this case $\boldsymbol{y} = \overline{\alpha} \cdots y_{j_2} y_{j_2+1} \cdots y_\ell, \widetilde{\boldsymbol{y}} = \alpha \cdots y_{j_2} \widetilde{y}_{j_2+1} \cdots \widetilde{y}_\ell$ where $y_i = \widetilde{y}_i = y_{j_2}$ for $i > j_2$. Moreover, in this case, it holds that

$$\alpha S, I = y_1 \cdots y_{j_2-1} B_{sub}\left(y_{j_2} \cdots y_\ell\right).$$

Hence, $|\alpha S, I| = \left|B_{sub}\left(y_{j_2} \cdots y_\ell\right)\right| \leqslant \ell - 2$ and according to Lemma 27, the following upper bounds hold

| $\lvert I, \alpha S\rvert$ | $\lvert \alpha S, I\rvert$ | $\lvert I, S\beta\rvert$ | $\lvert S\beta, I\rvert$ |
|---|---|---|---|
| 1 | $\ell - 2$ | 1 | 2 |

Furthermore, if $d_H(\boldsymbol{y}, \widetilde{\boldsymbol{y}}) \geqslant 3$, then $|I, I| \leqslant 2$ and $S, S = 0$, which means that

$$\begin{aligned} Sum \leqslant &2|I, I| + 2|S, S| + |I, \alpha S| + |\alpha S, I| \\ &+ |I, S\beta| + |S\beta, I| \leqslant \ell + 6. \end{aligned}$$

However, if $d_H(\boldsymbol{y}, \widetilde{\boldsymbol{y}}) \leqslant 2$, then at most one diagonal is maintained in section $[j_1, j_2]$ as this section

consists of at least 3 bits, i.e., $\boldsymbol{y} = \overline{\alpha}(\overline{\alpha})^h\alpha\alpha^t, \widetilde{\boldsymbol{y}} = \alpha(\overline{\alpha})^h\overline{\alpha}\alpha^t$ where $h, t \geqslant 1$. In this case, the following statements hold.

- $\alpha\overline{\alpha}(\overline{\alpha})^h\overline{\alpha}\alpha^t \in (\alpha S, \alpha S) \cap (\alpha S, I)$.
- If $\beta = \overline{\alpha}$, then $|S\beta, I| \leqslant 1$ and $|I, S\beta| \leqslant 1$.
- If $\beta = \alpha$, then $\overline{\alpha}(\overline{\alpha})^h\overline{\alpha}\alpha^t\alpha \in (S\beta, S\beta) \cap (I, S\beta)$.
- $I, I \leqslant 1$.
- $S, S \leqslant 2$.

Hence, the following upper bounds hold

| $\lvert I, I\rvert$ | $\lvert\alpha S, \alpha S \cup \alpha S, I\rvert$ | $\lvert I, \alpha S\rvert$ |
|---|---|---|
| 1 | $2 + \ell - 2 - 1$ | 1 |

If $\beta = \alpha$, then

| $\lvert I, I\rvert$ | $\lvert S\beta, S\beta \cup I, S\beta\rvert$ | $\lvert S\beta, I\rvert$ |
|---|---|---|
| 1 | $2 + 1 - 1$ | 2 |

Otherwise, $\beta = \overline{\alpha}$ and

| $\lvert I, I\rvert$ | $\lvert S\beta, S\beta\rvert$ | $\lvert I, S\beta\rvert$ | $\lvert S\beta, I\rvert$ |
|---|---|---|---|
| 1 | 2 | 1 | 1 |

Therefore, in both of the cases $\beta = \alpha$ and $\beta = \overline{\alpha}$, $Sum \leqslant \ell + 6$.

c) $j_1 > 1, j_2 = \ell$. The proof is similar to the case of $j_1 = 1, j_2 < \ell$.

d) $j_1 > 1, j_2 < \ell$. In this case, according to Proposition 30, at most one diagonal is maintained in each of the sections $[1, j_2]$, $[j_1, \ell]$, i.e., $c_1 = 1, c_2 = 1$. Moreover, if different diagonals are maintained in each of these sections, then according to Proposition 31, the diagonals are broken in $[j_1 - 1, j_1]$, $[j_2, j_2 + 1]$. Thus, the following upper bounds hold according to Lemma 27.

| $\lvert I, I\rvert$ | $\lvert S, S\rvert$ | $\lvert I, \alpha S\rvert + \lvert\alpha S, I\rvert$ | $\lvert I, S\beta\rvert + \lvert S\beta, I\rvert$ | $Sum$ |
|---|---|---|---|---|
| 2 | 2 | $2 + 1$ | $2 + 1$ | 14 |

Therefore, it is enough to consider words for which only one diagonal is maintained in all of the word. Assume without loss of generality that the diagonal $\boldsymbol{y}\searrow\widetilde{\boldsymbol{y}}$ is the one maintained in all of the word. Consider the following cases.

- $k = 1$. In this case the words $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ have one of the following forms.
  - $\boldsymbol{y} = \alpha^h\overline{\alpha}(\overline{\alpha})^t, \widetilde{\boldsymbol{y}} = \alpha^h\alpha(\overline{\alpha})^t$ where $h, t \geqslant 1$. Let

$$\begin{aligned} z_1 &= \alpha^h\alpha\overline{\alpha}(\overline{\alpha})^t, \\ z_2 &= \alpha^h\overline{\alpha}\alpha(\overline{\alpha})^t, \\ z_3 &= \alpha\alpha^h\alpha(\overline{\alpha})^t, \\ z_4 &= \alpha^h\overline{\alpha}(\overline{\alpha})^t\alpha, \\ z_5 &= \alpha^h\alpha(\overline{\alpha})^t\alpha, \\ z_6 &= \alpha^h\overline{\alpha}(\overline{\alpha})^t\overline{\alpha}. \end{aligned}$$

Then,

$$\begin{aligned} &I, I = \{z_1, z_2\}, \\ &\alpha S, \alpha S = \{z_1, z_3\}, \\ &I, \alpha S = \{z_1, z_2\}, \\ &\alpha S, I = \alpha\alpha^h B_{sub}\left(\overline{\alpha}(\overline{\alpha})^t\right). \end{aligned}$$

If $\beta = \alpha$, then

$$\begin{aligned} &S\beta, S\beta = \{z_4, z_5\}, \\ &S\beta, I = \{z_5\}, \\ &I, S\beta = \{z_4\}. \end{aligned}$$

Otherwise,

$$\begin{aligned} &S\beta, S\beta = \{z_1, z_6\}, \\ &S\beta, I = \{z_1, z_2\}, \\ &I, S\beta = B_{sub}\left(\alpha^h\alpha\right)(\overline{\alpha})^t\overline{\alpha}. \end{aligned}$$

Since

$$\begin{aligned} &z_3 \in \alpha S, I, \\ &|\alpha S, I| = \left|B_{sub}\left(\overline{\alpha}(\overline{\alpha})^t\right)\right| \leqslant \ell - 1, \end{aligned}$$

and since for $\beta = \overline{\alpha}, z_6 \in I, S\beta$, and

$$\begin{aligned} |\alpha S, I| + |I, S\beta| &= \left|B_{sub}\left(\overline{\alpha}(\overline{\alpha})^t\right)\right| \\ &\quad + \left|B_{sub}\left(\alpha^h\alpha\right)\right| = \ell + 1, \end{aligned}$$

then

$$\begin{aligned} &Sum \leqslant \ell + 5, \beta = \alpha, \\ &Sum = \ell + 5, \beta = \overline{\alpha}. \end{aligned}$$

  - $\boldsymbol{y} = (\overline{\alpha})^h\alpha\alpha^t, \widetilde{\boldsymbol{y}} = (\overline{\alpha})^h\overline{\alpha}\alpha^t$ where $h, t \geqslant 1$. In this case, if $\beta = \alpha$, then the solution is identical to the previous case when $\beta = \alpha$. Otherwise, $\beta = \overline{\alpha}$ and according to Proposition 32, $Sum \leqslant 12$.
- $k = 2$. In this case the words $\boldsymbol{y}, \widetilde{\boldsymbol{y}}$ have one of the following forms.
  - $\boldsymbol{y} = \alpha^h\overline{\alpha}\alpha\alpha^t, \widetilde{\boldsymbol{y}} = \alpha^h\alpha\overline{\alpha}\alpha^t$. Let

$$\begin{aligned} z_1 &= \alpha^h\alpha\overline{\alpha}\alpha\alpha^t, \\ z_2 &= \alpha^h\overline{\alpha}\alpha\overline{\alpha}\alpha^t, \\ z_3 &= \alpha\alpha^h\alpha\alpha\alpha^t, \\ z_4 &= \alpha\alpha^h\overline{\alpha\alpha}\alpha^t, \\ z_5 &= \alpha^h\overline{\alpha\alpha}\alpha^t\overline{\alpha}, \\ z_6 &= \alpha^h\alpha\alpha\alpha^t\overline{\alpha}, \\ z_7 &= \alpha^h\overline{\alpha\alpha}\alpha^t\alpha. \end{aligned}$$

Then,

$$\begin{aligned} &I, I = \{z_1, z_2\}, \\ &\alpha S, \alpha S = \{z_3, z_4\}, \\ &\alpha S, I = \alpha\alpha^h\overline{\alpha}B_{sub}\left(\alpha\alpha^t\right), \\ &I, \alpha S = \{z_2\}. \end{aligned}$$

If $\beta = \overline{\alpha}$, then

$$\begin{aligned} &S\beta, S\beta = \{z_5, z_6\}, \\ &S\beta, I = I, S\beta = \emptyset. \end{aligned}$$

Otherwise,

$$S\beta, S\beta = \{z_3, z_7\},$$
$$S\beta, I = \{z_2\},$$
$$I, S\beta = B_{sub}\left(\alpha^h\alpha\right)\overline{\alpha}\alpha^t\alpha.$$

Since $z_4 \in \alpha S, I$,

$$|\alpha S, I| = \left|B_{sub}\left(\alpha\alpha^t\right)\right| \leqslant \ell - 2,$$

and since for $\beta = \alpha$, $z_7 \in I, S\beta$,

$$|\alpha S, I| + |I, S\beta| = \left|B_{sub}\left(\overline{\alpha}\alpha^t\right)\right| + \left|B_{sub}\left(\alpha^h\alpha\right)\right| = \ell,$$

then

$$Sum = \ell + 6, \beta = \alpha,$$
$$Sum \leqslant \ell + 5, \beta = \overline{\alpha}.$$

- $\boldsymbol{y} = (\overline{\alpha})^h\alpha\overline{\alpha}(\overline{\alpha})^t, \widetilde{\boldsymbol{y}} = (\overline{\alpha})^h\overline{\alpha}\alpha(\overline{\alpha})^t$ where $h, t \geqslant 1$. In this case, if $\beta = \overline{\alpha}$, then the solution is identical to the previous case when $\beta = \overline{\alpha}$. Otherwise, $\beta = \alpha$ and according to Proposition 32, $Sum \leqslant 12$.

- $k \geqslant 3$. In this case, if $d_H(\boldsymbol{y}, \widetilde{\boldsymbol{y}}) \geqslant 3$, then $|I, I| \leqslant 2, |S, S| = 0$. Hence,

$$2|I, I| + |\alpha S, \alpha S| + |S\beta, S\beta| \leqslant 4.$$

However, if $d_H(\boldsymbol{y}, \widetilde{\boldsymbol{y}}) \leqslant 2$, then $|I, I| \leqslant 1$, $|S, S| \leqslant 2$. Hence,

$$2|I, I| + |\alpha S, \alpha S| + |S\beta, S\beta| \leqslant 6,$$

where this bound can be achieved only if exactly one diagonal is maintained in $[j_1, j_2]$. Furthermore, note that since $j_1 > 1$, $j_2 < \ell$ and the diagonal $\boldsymbol{y}\searrow\widetilde{\boldsymbol{y}}$ is the only one maintained in all of the word, then $y_1 = \widetilde{y}_{j_1}$ and $y_\ell = y_{j_2}$. Next consider the following cases.

- $y_1 = \overline{\alpha}, y_\ell = \overline{\beta}$. In this case, according to Proposition 32, $Sum \leqslant 12$.
- $y_1 = \overline{\alpha}, y_\ell = \beta$. In this case, the following upper bounds hold

| $2\|I, I\| + 2\|S, S\|$ | $\|I, \alpha S\|$ | $\|\alpha S, I\|$ | $\|I, S\beta\|$ | $\|S\beta, I\|$ | $Sum$ |
|---|---|---|---|---|---|
| 6 | 0 | 0 | $\ell - 2$ | 2 | $\ell + 6$ |

where the equalities $|I, \alpha S| = |\alpha S, I| = 0$ can be proved as follows. The words in $I, \alpha S$ begin with $\alpha$, and hence can be received from $\boldsymbol{y}$ only by an insertion of $\alpha$ at the beginning of $\boldsymbol{y}$. However, such an insertion leads to a word of Hamming distance 2 from $\alpha\widetilde{\boldsymbol{y}}$, i.e., a word that is not in $\alpha B_{sub}(\widetilde{\boldsymbol{y}})$.
- $y_1 = \alpha, y_\ell = \overline{\beta}$. The proof is similar to the previous case.
- $y_1 = \alpha, y_\ell = \beta$. In this case, observe that the following holds.
  - $y_i = \widetilde{y}_i = \alpha, i < j_1$.
  - $y_{j_1} = \overline{\alpha}, \widetilde{y}_{j_1} = \alpha$.
  - $y_{j_2} = \beta, \widetilde{y}_{j_2} = \overline{\beta}$.
  - $y_i = \widetilde{y}_i = \beta, i > j_2$.
  - $|I, \alpha S| \leqslant 1$. This claim can be proved by observing that any word $\boldsymbol{z} \in I, \alpha S$ satisfies $z_{j_1} = \overline{\alpha}$ and there is only one such word in $\alpha B_{sub}(\widetilde{\boldsymbol{y}})$. Assume to the contrary that there is a word $\boldsymbol{x} \in I, \alpha S$ such that $x_{j_1} = \alpha$. Such a word can be received only by insertion of $\alpha$ at the beginning of $\boldsymbol{y}$. Hence $\boldsymbol{x} = \alpha\boldsymbol{y}$. However, $\boldsymbol{x} \notin \alpha B_{sub}(\widetilde{\boldsymbol{y}})$ as $d_H(\alpha\boldsymbol{y}, \alpha\widetilde{\boldsymbol{y}}) = d_H(\boldsymbol{y}, \widetilde{\boldsymbol{y}}) \geqslant 2$.
  - $|S\beta, I| \leqslant 1$, same proof as $|I, \alpha S| \leqslant 1$.
  - $|\alpha S, I| + |I, S\beta| \leqslant \ell - 1$.

  Considering all of these claims, the following upper bounds can be derived.

| $2\|I, I\| + 2\|S, S\|$ | $\|I, \alpha S\|$ | $\|\alpha S, I\| + \|I, S\beta\|$ | $\|S\beta, I\|$ | $Sum$ |
|---|---|---|---|---|
| 6 | 1 | $\ell - 1$ | 1 | $\ell + 7$ |

  Observe that this upper bound may be achieved only if the following conditions hold.
  - $d_H(\boldsymbol{y}, \widetilde{\boldsymbol{y}}) = 2$ and exactly one diagonal is maintained in $[j_1, j_2]$. Otherwise, $|I, I| + |\alpha S, \alpha S| + |S\beta, S\beta| < 6$.
  - $j_1 + 2 = j_2$. Otherwise, $|\alpha S, I| + |I, S\beta| < \ell - 1$.

  Therefore, it is enough to consider words satisfying these two properties. Such words are $\boldsymbol{y} = \alpha^h\alpha\overline{\alpha}\alpha\alpha^t, \widetilde{\boldsymbol{y}} = \alpha^h\alpha\overline{\alpha}\overline{\alpha}\alpha^t$ where $h, t \geqslant 1$. In this case, the word $\alpha\alpha^h\alpha\overline{\alpha}\overline{\alpha}\alpha^t$ is counted twice, once in $\alpha S, \alpha S$, and once in $\alpha S, I$. Therefore, the upper bound $\ell + 7$ cannot be achieved for these words too. ■

## REFERENCES


[1] M. A. Sini and E. Yaakobi, "Reconstruction of sequences in DNA storage," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Paris, France, Jul. 2019, pp. 290–294.

[2] D. Bar-Lev, T. Etzion, and E. Yaakobi, "On the size of Levenshtein balls," in *Proc. Int. Symp. Inf. Theory*, Melbourne, VIC, Australia, Jul. 2021, pp. 1979–1984.

[3] D. Carmean, L. Ceze, G. Seelig, K. Stewart, K. Strauss, and M. Willsey, "DNA data storage and hybrid molecular-electronic computing," *Proc. IEEE*, vol. 107, no. 1, pp. 63–72, Jan. 2019.

[4] R. P. Feynman, "There's plenty of room at the bottom," *Eng. Sci.*, vol. 23, no. 5, pp. 22–36, Feb. 1960.

[5] R. Gabrys and E. Yaakobi, "Sequence reconstruction over the deletion channel," *IEEE Trans. Inf. Theory*, vol. 64, no. 4, pp. 2924–2931, Apr. 2018.

[6] R. Heckel, G. Mikutis, and R. N. Grass, "A characterization of the DNA data storage channel," *Sci. Rep.*, vol. 9, no. 1, Dec. 2019, Art. no. 9663.

[7] D. S. Hirschberg, "Bounds on the number of string subsequences," in *Proc. Annu. Symp. Combinat. Pattern Matching*, 1999, pp. 115–122.

[8] M. Horovitz and E. Yaakobi, "Reconstruction of sequences over non-identical channels," *IEEE Trans. Inf. Theory*, vol. 65, no. 2, pp. 1267–1286, Feb. 2019.

[9] T. Jiang and A. Vardy, "Asymptotic improvement of the Gilbert–Varshamov bound on the size of binary codes," *IEEE Trans. Inf. Theory*, vol. 50, no. 8, pp. 1655–1664, Aug. 2004.

[10] V. Junnila, T. Laihonen, and T. Lehtilä, "On Levenshtein's channel and list size in information retrieval," *IEEE Trans. Inf. Theory*, vol. 67, no. 6, pp. 3322–3341, Aug. 2020.

[11] E. Konstantinova, "Reconstruction of permutations distorted by reversal errors," *Discrete Appl. Math.*, vol. 155, no. 18, pp. 2426–2434, Nov. 2007.

[12] E. Konstantinova, "On reconstruction of signed permutations distorted by reversal errors," *Discrete Math.*, vol. 308, pp. 974–984, Mar. 2008.

[13] E. Konstantinova, V. Levenshtein, and J. Siemons, "Reconstruction of permutations distorted by single transposition errors," Feb. 2007, *arXiv:math/0702191*. [Online]. Available: http://arxiv.org/abs/math/0702191v1
[14] S. Kosuri and G. M. Church, "Large-scale de novo DNA synthesis: Technologies and applications," *Nature Methods*, vol. 11, no. 5, pp. 499–507, May 2014.
[15] V. I. Levenshtein, "Efficient reconstruction of sequences," *IEEE Trans. Inf. Theory*, vol. 47, no. 1, pp. 2–22, Jan. 2001.
[16] V. I. Levenshtein, "Efficient reconstruction of sequences from their subsequences or supersequences," *J. Combinat. Theory A*, vol. 93, no. 2, pp. 310–332, 2001.
[17] V. I. Levenshtein, E. Konstantinova, E. Konstantinov, and S. Molodtsov, "Reconstruction of a graph from 2-vicinities of its vertices," *Discrete Appl. Math.*, vol. 156, pp. 1399–1406, May 2008.
[18] V. I. Levenshtein and J. Siemons, "Error graphs and the reconstruction of elements in groups," *J. Combinat. Theory, A*, vol. 116, no. 4, pp. 795–815, 2009.
[19] L. Organick *et al.*, "Random access in large-scale DNA data storage," *Nature Biotechnol.*, vol. 36, no. 3, pp. 242–248, 2018.
[20] F. Sala and L. Dolecek, "Counting sequences obtained from the synchronization channel," in *Proc. IEEE Int. Symp. Inf. Theory*, Istanbul, Turkey, Jul. 2013, pp. 2925–2929.
[21] F. Sala, R. Gabrys, C. Schoeny, and L. Dolecek, "Exact reconstruction from insertions in synchronization codes," *IEEE Trans. Inf. Theory*, vol. 63, no. 4, pp. 2428–2445, Apr. 2017.
[22] E. Yaakobi and J. Bruck, "On the uncertainty of information retrieval in associative memories," *IEEE Trans. Inf. Theory*, vol. 65, no. 4, pp. 2155–2165, Apr. 2019.
[23] E. Yaakobi, M. Schwartz, M. Langberg, and J. Bruck, "Sequence reconstruction for Grassmann graphs and permutations," in *Proc. Int. Symp. Inform. Theory*, Istanbul, Turkey, Jul. 2013, pp. 874–878.

**Maria Abu-Sini** (Student Member, IEEE) received the B.A. and M.Sc. degrees in computer science from the Technion—Israel Institute of Technology, Haifa, Israel, in 2017 and 2020, respectively, where she is currently pursuing the Ph.D. degree with the Computer Science Department. Her research interests include algorithms, information theory, and coding theory with applications to DNA-based storage.

**Eitan Yaakobi** (Senior Member, IEEE) received the B.A. degree 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 at San Diego, in 2011. From 2011 to 2013, he was a Post-Doctoral Researcher with the Department of Electrical Engineering, California Institute of Technology, and the Center for Memory and Recording Research, University of California at San Diego. He is currently an Associate Professor with the Computer Science Department, Technion—Israel Institute of Technology. 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.