IEICE TRANSACTIONS

on Fundamentals of Electronics, Communications and Computer Sciences

DOI:10.1587/transfun.2025EAP1078

Publicized:2025/10/31

This article has been accepted and published on J-STAGE in advance of copyediting. Content is final as presented.

A PUBLICATION OF THE ENGINEERING SCIENCES SOCIETY

The Institute of Electronics, Information and Communication Engineers

Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER *Special Section on Information Theory and Its Applications*

One Bit-Flipping/Insertion/Deletion Correcting Methods for Substrings of Binary Circular String*

Takahiro OTA†, ***Senior Member***, **Keita KAMIYA**†, ***Nonmember***, ***and*** **Akiko MANADA**††, ***Member***

SUMMARY Compression by Substring Enumeration (CSE), which is one of the lossless data compression algorithms, and various versions of CSE have been proposed. In encoding of CSE, substrings of given fixed length and their frequencies within circular string for an input string are output as a codeword. The circular string is made by connecting the first symbol and the last symbol of an input string. In decoding of CSE, the circular string is reconstructed from its substrings and their frequencies. Furthermore, the minimum length of substrings for which the decoding does reconstruct the circular string has been proved, together with a reconstruction algorithm. However, the algorithm requires substrings to have no errors.

Therefore, in this paper, we propose an error correcting algorithm which can detect one of the substrings having one bit-flipping, one bit-insertion, or one bit-deletion error and correct the bit error. By applying the proposed algorithm, we can reconstruct a circular string from a set of substrings and their frequencies including only one substring which has at most one bit error.

key words: *circular string, substring with frequency, error correcting, reconstruction*

1. Introduction

Reconstruction of an input string from its substrings of given fixed length is useful for lossless data compression, deoxyribonucleic acid (DNA) storage system, and so on [2, 3]. For an input string, Compression by Substring Enumeration (CSE) [2], which is one of the lossless data compression algorithms, outputs a set of substrings of given fixed length and their frequencies in the circular string of the input string as codeword. Various versions of CSE have been proposed [4–7] and it is shown that performance of CSE in [4] is better than that of well-known lossless data compression algorithm applications such as bzip2 [8]. In decoding of CSE, a reconstruction algorithm assembles the circular string of an input string from its substrings of fixed length and their frequencies.

Moreover, it is proved on the minimum length of substrings for a given binary circular string for reconstructing the circular string from its substrings and their frequencies, and an efficient reconstruction algorithm from substrings of the length is proposed [9]. However, the algorithm requires substrings without errors for reconstruction of the circular string.

In this paper, we propose an error correcting algorithm which can detect one of the substrings having either one bit-flipping, bit-insertion, or bit-deletion and correct the error bit. By applying the proposed algorithm, a reconstruction algorithm [9] can assemble a circular string from a set of substrings of fixed length and their frequencies including only one substring which has at most one bit error.

2. Notations and Definitions

Let Σ be a binary source alphabet $\Sigma = \{0, 1\}$ and $\bar{a} = 1 - a$ for $a \in \Sigma$. We denote by Σ^n the set of all strings of length n over Σ, and define the set of all finite strings over Σ to be $\Sigma^* = \cup_{n\geq 0}\Sigma^n$, including the *empty string* ϵ of length 0.

Let $|\boldsymbol{x}|$ be the length of $\boldsymbol{x} \in \Sigma^*$, e.g. $|\boldsymbol{x}| = n$ for $\boldsymbol{x}(= x_1 \dots x_n) \in \Sigma^n$. For $\boldsymbol{x} \in \Sigma^*$, $\boldsymbol{x}$ can be written as $\boldsymbol{x} = \boldsymbol{uvw}$ in terms of a concatenation of strings $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w} \in \Sigma^*$. In this case, $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{w}$ are called a *prefix*, *substring*, and *suffix* of $\boldsymbol{x}$, respectively. For $a, b \in \Sigma$ and $\boldsymbol{y} \in \Sigma^*$, let $\pi(\boldsymbol{y}a)$ and $\sigma(b\boldsymbol{y})$ be $\pi(\boldsymbol{y}a) = \boldsymbol{y}$ and $\sigma(b\boldsymbol{y}) = \boldsymbol{y}$, respectively. We also use a^r for a positive integer r to indicate the string generated by repeating $a \in \Sigma$ r times.

For a given $\boldsymbol{x} = x_1x_2 \dots x_n \in \Sigma^n$ and $n \geq 1$, the string obtained by circularly concatenating the last symbol x_n and the first symbol x_1 is called the *circular string* $\tilde{\boldsymbol{x}}$ of $\boldsymbol{x}$. For convenience, we always set n to be the length of $\boldsymbol{x}$ and we assume $n \geq 3$ throughout this paper.

For positive integer k $(1 \leq k \leq n)$ and $\tilde{\boldsymbol{x}}$, let $\mathcal{S}_k(\tilde{\boldsymbol{x}})$ be a multiset of substrings of length k in $\tilde{\boldsymbol{x}}$, that is

$$\mathcal{S}_k(\tilde{\boldsymbol{x}}) = [x_1 \dots x_k, x_2 \dots x_{k+1}, \dots, x_n \dots x_{k-1}]. \quad (1)$$

Note that $[\cdot]$ denotes a multiset and the size of $\mathcal{S}_k(\tilde{\boldsymbol{x}})$ is always n. For example, for $\boldsymbol{x} = 1101$ and $k = 2$,

$$\mathcal{S}_2(\tilde{\boldsymbol{x}}) = [01, 10, 11, 11]. \quad (2)$$

For a string $\boldsymbol{w}$ of length k $(1 \leq k \leq n)$, let $C(\boldsymbol{w})$ be the number of occurrences of $\boldsymbol{w}$ in $\mathcal{S}_k(\tilde{\boldsymbol{x}})$. For example, based on (2), we have $C(00) = 0, C(01) = C(10) = 1, C(11) = 2$.

We define $C_\sigma(\boldsymbol{v})$ and $C_\pi(\boldsymbol{v})$ for $\boldsymbol{v} \in \Sigma^{k-1}$ as follows.

$$C_\sigma(\boldsymbol{v}) := \sum_{a\in\Sigma} C(a\boldsymbol{v}), \quad (3)$$

$$C_\pi(\boldsymbol{v}) := \sum_{b\in\Sigma} C(\boldsymbol{v}b). \quad (4)$$

†The author is with the School of Network and Information, Senshu University, 2-1-1, Higashimita, Tama, Kawasaki, Kanagawa, 214-8580, JAPAN, Email:ota@isc.senshu-u.ac.jp.

††The author is with the Dept. of Electrical, Electronics and Information Engineering, Nagaoka University of Technology, 1603-1, Kamitomioka, Nagaoka, Niigata, 940-2188, JAPAN, Email:amanada@vos.nagaokaut.ac.jp.

*This paper was presented in part at [1], in Proc. of ISITA2024, November 2024.

Copyright © 200x The Institute of Electronics, Information and Communication Engineers

For a positive integer k $(1 \leq k \leq n)$ and $\boldsymbol{v} \in \Sigma^{k-1}$, observe that

$$\sum_{\boldsymbol{w} \in \Sigma^k} C(\boldsymbol{w}) = n, \tag{5}$$

$$C_\sigma(\boldsymbol{v}) = C_\pi(\boldsymbol{v}). \tag{6}$$

We use the following Assumption 1 throughout this paper.

Assumption 1: Any two substrings of $\mathcal{S}_n(\tilde{\boldsymbol{x}})$ are distinct.

Note that Assumption 1 is natural because $\tilde{\boldsymbol{x}}$ can be constructed from a shortest periodic string $\boldsymbol{x}$ of length n within $\tilde{\boldsymbol{x}}$.

3. Conventional Method

The conventional algorithms that reconstruct the input string from the multiset of its proper substrings are classified into two types. The first algorithm reconstructs the same string as an input string under a noiseless channel, and is used for applications such as lossless data compression [2]. The second algorithm reconstructs the almost the same string as an input string under a noisy channel, and is used for applications such as DNA sequencing [3]. For each algorithm, additional information, such as parity bits and unique identifiers of the substrings [10], will be necessary for extension: in the first case, extending from a noiseless channel to a noisy channel, and in the second case, from an almost the same string to the exact string. However, to the best of our knowledge, any reconstructing algorithm under a noisy channel without additional information has not been addressed so far. In this paper, we propose such a reconstruction algorithm based on the algorithm [9] explained below, which works under noiseless channel.

Reconstruction without Errors

The algorithm in [9] shows that $\tilde{\boldsymbol{x}}$ can be reconstructed from $\mathcal{S}_k(\tilde{\boldsymbol{x}})$ for given proper length k. It also shows the minimum length k_0 of substrings for reconstructing $\tilde{\boldsymbol{x}}$ from $\mathcal{S}_{k_0}(\tilde{\boldsymbol{x}})$ and an efficient reconstruction algorithm from $\mathcal{S}_{k_0}(\tilde{\boldsymbol{x}})$. However, since the condition of length k is slightly complicated, we use a simplified assumption of length k of substrings in this paper, as denoted below.

Assumption 2: The length k of the substring satisfies the following two conditions;

- (a) For each substring $\boldsymbol{v}a \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ of length k, $\boldsymbol{v}\bar{a} \notin \mathcal{S}_k(\tilde{\boldsymbol{x}})$ except for at most one pair of substrings $\boldsymbol{w}b \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ and $\boldsymbol{w}\bar{b} \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ of length k; and
- (b) If an exceptional pair of substrings exists, then $C(\boldsymbol{w}b) = 1$ and $C(\boldsymbol{w}\bar{b}) \geq 1$ for some $b \in \Sigma$.

For any $\tilde{\boldsymbol{x}}$ satisfying Assumption 1, there exists k $(\leq n)$ in Assumption 2, where $n \geq 3$ [9]. Note that any k shown in Assumption 2 is longer than the minimum length k_0 [9] for some $\tilde{\boldsymbol{x}}$, and Assumption 2 is equivalent to Sufficient Conditions A and B shown in [5].

For example, for $\boldsymbol{x} = 1010101000$ of length 10, $k(=4)$ satisfies Assumption 2, and

$$\mathcal{S}_4(\tilde{\boldsymbol{x}}) = [0001, 0010, 0100, 0101, 0101, 0101, 1000, 1010, 1010, 1010]. \tag{7}$$

In (7), an exceptional pair of substrings is given by 0100 and 0101 where $C(0100) = 1$ and $C(0101) = 3$. The pair satisfies (b) of Assumption 2. The algorithm can be reconstructed $\tilde{\boldsymbol{x}}$ for $\boldsymbol{x} = 1010101000$ from $\mathcal{S}_4(\tilde{\boldsymbol{x}})$ in (7).

However, the algorithm [9] requires substrings without errors for reconstruction of the circular string.

4. Proposed Method

In this section, we propose an error correction algorithm when only one substring of $\mathcal{S}_k(\tilde{\boldsymbol{x}})$ suffers an error. More precisely, we consider the case when one bit-flipping, one bit-insertion, or one bit-deletion error occurs within a substring. We assume $k \geq 2$ for k to satisfy Assumption 2.

Reconstruction with Errors

For $\boldsymbol{u} \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$, let $\bar{\boldsymbol{u}}$ be a substring generated by one bit-flipping, one bit-insertion, or one bit-deletion error from $\boldsymbol{u}$. If $|\bar{\boldsymbol{u}}| = k + 1$ (resp. $|\bar{\boldsymbol{u}}| = k - 1$), then $\bar{\boldsymbol{u}}$ has an one bit-insertion (resp. deletion) error. If $|\bar{\boldsymbol{u}}| = k$, then $\bar{u} = u_1 \ldots \bar{u}_i \ldots u_k$ when a flipping error occurs in the i-th bit in $\boldsymbol{u}$.

Let $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ be a multiset $\mathcal{S}_k(\tilde{\boldsymbol{x}})$ which has $\bar{\boldsymbol{u}}$ instead of one substring $\boldsymbol{u}$, that is

$$\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}}) = (\mathcal{S}_k(\tilde{\boldsymbol{x}}) \backslash [\boldsymbol{u}]) \cup [\bar{\boldsymbol{u}}]. \tag{8}$$

In other words, $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ is obtained by removing one of $\boldsymbol{u}$ from multiset $\mathcal{S}_k(\tilde{\boldsymbol{x}})$ and by adding $\bar{\boldsymbol{u}}$ to the multiset. Therefore, the sizes of $\mathcal{S}_k(\tilde{\boldsymbol{x}})$ and $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ are equal to n.

For example, when either one bit-insertion or one bit-deletion error occurs in (7), $\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})$ are respectively given by

$$\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}}) = [0001, 0010, \mathbf{01001}, 0101, 0101, 0101, 1000, 1010, 1010, 1010], \tag{9}$$

$$\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}}) = [0001, 0010, 0100, 0101, \mathbf{011}, 0101, 1000, 1010, 1010, 1010]. \tag{10}$$

In (9), $\bar{\boldsymbol{u}} = 01001$ and $\boldsymbol{u} = 0100$. In (10), $\bar{\boldsymbol{u}} = 011$ and $\boldsymbol{u} = 0101$. Since $n \geq 3$ holds, we can know insertion and deletion errors by majority decision with respect to lengths of substrings in $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$.

Similarly, when one bit-flipping error occurs in (7), $\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})$ is given by

$$\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}}) = [0001, 0010, \mathbf{0101}, 0101, 0101, 0101, 1000, 1010, 1010, 1010]. \tag{11}$$

In (11), $\bar{\boldsymbol{u}} = 0101$ and $\boldsymbol{u} = 0100$.

Let $\overline{C}(\boldsymbol{w})$ be the number of occurrences of $\boldsymbol{w}$ in $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$. We define $\overline{C}_\sigma(\boldsymbol{v})$ and $\overline{C}_\pi(\boldsymbol{v})$ for $\boldsymbol{v} \in \Sigma^{k-1}$ as follows.

$$\overline{C}_\sigma(\boldsymbol{v}) := \sum_{a\in\Sigma} \overline{C}(a\boldsymbol{v}), \tag{12}$$

$$\overline{C}_\pi(\boldsymbol{v}) := \sum_{b\in\Sigma} \overline{C}(\boldsymbol{v}b). \tag{13}$$

There exists some $\boldsymbol{v}$ such that $\overline{C}_\sigma(\boldsymbol{v}) \neq \overline{C}_\pi(\boldsymbol{v})$ because $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ has $\bar{\boldsymbol{u}}$ instead of $\boldsymbol{u}$. For example, $\overline{C}(0101) = 4$ and $\overline{C}(1101) = 0$ in (11), so that $\overline{C}_\sigma(101) = 4$. On the other hand, $\overline{C}(1010) = 3$ and $\overline{C}(1011) = 0$ in (11), so that $\overline{C}_\pi(101) = 3$. Therefore, for $\boldsymbol{v} = 101$, $\overline{C}_\sigma(101) \neq \overline{C}_\pi(101)$.

As for insertion and deletion errors, detection and correction scheme of the errors will be described in Sec.4.1. As for flipping error, detection and correction schemes of the error will be described in Sec.4.2.

4.1 Error Correction of Insertion or Deletion Error

As for $\bar{\boldsymbol{u}}$, we can find $\bar{\boldsymbol{u}}$ by length of $|\bar{\boldsymbol{u}}|$ because $|\bar{\boldsymbol{u}}| = k+1$ or $|\bar{\boldsymbol{u}}| = k-1$ and $n \geq 3$. Therefore, we know $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ and $\bar{\boldsymbol{u}}$ in advance when either an insertion or a deletion error occurs. To avoid using substrings of $\bar{\boldsymbol{u}}$ as substrings of $\boldsymbol{u}$, we set $\overline{C}(\bar{\boldsymbol{u}}) = 0$. It implies that we can use $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})\backslash[\bar{\boldsymbol{u}}]$ instead of $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$.

First, we prove Lemma 1 which determines whether $\boldsymbol{u}$ is a string of single letter or not.

Lemma 1: For $a, b \in \Sigma$ and $\bar{\boldsymbol{u}}$ such that $|\bar{\boldsymbol{u}}| = |\boldsymbol{u}| + 1 (= k+1)$ or $|\bar{\boldsymbol{u}}| = |\boldsymbol{u}| - 1 (= k-1)$, we have

$$\overline{C}_\sigma(\sigma(\boldsymbol{u})) = \overline{C}(\sigma(\boldsymbol{u})a) + \overline{C}(\sigma(\boldsymbol{u})\bar{a}) - \alpha, \tag{14}$$

$$\overline{C}_\pi(\pi(\boldsymbol{u})) = \overline{C}(b\pi(\boldsymbol{u})) + \overline{C}(\bar{b}\pi(\boldsymbol{u})) - \alpha, \tag{15}$$

where $\alpha = 0$ when $\boldsymbol{u} = c^k$ $(c \in \Sigma)$, and $\alpha = 1$ otherwise.

Proof.

We first suppose that $\boldsymbol{u}$ is a string consisting of two kinds of letters. We will show that $\boldsymbol{u}$ satisfies (14). Since $\boldsymbol{u}$ has two kinds of letters, $\pi(\boldsymbol{u}) \neq \sigma(\boldsymbol{u})$. Therefore, $\boldsymbol{u} \neq \sigma(\boldsymbol{u})d$ and $\boldsymbol{u} \neq \sigma(\boldsymbol{u})\bar{d}$ are satisfied, so that $(\mathcal{S}_k(\tilde{\boldsymbol{x}})\backslash[\boldsymbol{u}]) = (\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})\backslash[\bar{\boldsymbol{u}}])$. Hence,

$$C(\sigma(\boldsymbol{u})d) + C(\sigma(\boldsymbol{u})\bar{d}) = \overline{C}(\sigma(\boldsymbol{u})d) + \overline{C}(\sigma(\boldsymbol{u})\bar{d}) \tag{16}$$

holds. Since $(\mathcal{S}_k(\tilde{\boldsymbol{x}})\backslash[\boldsymbol{u}]) = (\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})\backslash[\bar{\boldsymbol{u}}])$,

$$C_\sigma(\sigma(\boldsymbol{u})) = \overline{C}_\sigma(\sigma(\boldsymbol{u})) + 1 \tag{17}$$

also holds. From (6),

$$C_\pi(\sigma(\boldsymbol{u})) = C_\sigma(\sigma(\boldsymbol{u})). \tag{18}$$

From (16), (17), and (18), we have (14) for $\alpha = 1$. Since $\tilde{\boldsymbol{x}}$ is a circular string, (15) also holds when $\alpha = 1$ from the symmetry of $\tilde{\boldsymbol{x}}$. Therefore, (14) and (15) hold when $\boldsymbol{u}$ is a string consisting of two kinds of letters.

Next, we suppose that $\boldsymbol{u}$ is a string consisting of single letter. Since $\pi(\boldsymbol{u}) = \sigma(\boldsymbol{u})$, $\boldsymbol{u}$ can be written by $\sigma(\boldsymbol{u})d$ for some $d \in \Sigma$. Therefore,

$$C(\sigma(\boldsymbol{u})d) + C(\sigma(\boldsymbol{u})\bar{d}) = \overline{C}(\sigma(\boldsymbol{u})d) + \overline{C}(\sigma(\boldsymbol{u})\bar{d}) + 1 \tag{19}$$

holds. From (17), (18), and (19), we have (14) for $\alpha = 0$. Since $\tilde{\boldsymbol{x}}$ is a circular string, (15) also holds when $\alpha = 0$ from the symmetry of $\tilde{\boldsymbol{x}}$. Therefore, (14) and (15) hold when $\boldsymbol{u}$ is a string consisting of single letter. It completes the proof. □

Note that symbols $a', b' \in \Sigma$ satisfying $\sigma(\boldsymbol{u})a' \in \overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})\backslash[\bar{\boldsymbol{u}}]$ and $b'\pi(\boldsymbol{u}) \in \overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})\backslash[\bar{\boldsymbol{u}}]$, respectively, do exist.

We next prove Proposition 1 which determines $\boldsymbol{u}$ from $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$.

Proposition 1: For $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ such that $\bar{\boldsymbol{u}}$ has either one bit-insertion error or one bit-deletion error, we can determine $\bar{\boldsymbol{u}}$ and correct $\bar{\boldsymbol{u}}$ to $\boldsymbol{u}$.

Proof. As for $\bar{\boldsymbol{u}}$, we can find $\bar{\boldsymbol{u}}$ by length of $|\bar{\boldsymbol{u}}|$ because $|\bar{\boldsymbol{u}}| = k+1$ or $|\bar{\boldsymbol{u}}| = k-1$ and $n \geq 3$. Moreover, we can know length k of $\boldsymbol{u}$ by majority decision because $n \geq 3$. From Lemma 1, by using values of α, $\boldsymbol{u}$ can be classified into two cases

(a) $\boldsymbol{u}$ is a string including two kinds of letters, or
(b) $\boldsymbol{u}$ is a string including only single letter,

where (a) and (b) are satisfied when $\alpha = 1$ and $\alpha = 0$, respectively.

We first consider case (a). From (14) and (15), we can know $\sigma(\boldsymbol{u})$ and $\pi(\boldsymbol{u})$ when $\alpha = 1$. Therefore, $\boldsymbol{u}$ is given by

$$\boldsymbol{u} = c\pi(\sigma(\boldsymbol{u}))d, \tag{20}$$

where $c \in \Sigma$ is the first symbol of $\pi(\boldsymbol{u})$ and $d \in \Sigma$ is the last symbol of $\sigma(\boldsymbol{u})$.

We next consider case (b). Recall that case (b) occurs when $\alpha = 0$. First, we consider an insertion error. In this case, $\bar{\boldsymbol{u}}$ is a string of single letter or a string including only one different letter in $k+1 (\geq 3)$ symbols of $\bar{\boldsymbol{u}}$. Therefore, symbol e is the majority symbol in $\bar{\boldsymbol{u}}$, and $\boldsymbol{u}$ is given by

$$\boldsymbol{u} = e^k \tag{21}$$

because $k \geq 2$.

Next, we consider a deletion error. In this case, $\bar{\boldsymbol{u}}$ is a string of single letter f, and $\boldsymbol{u}$ is given by

$$\boldsymbol{u} = f^k. \tag{22}$$

It completes the proof. □

For example, we show an example of insertion error, that is

$$\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}}) = [0001, 0010, \mathbf{01001}, 0101, 0101, 0101, 1000, 1010, 1010, 1010]. \tag{23}$$

For $\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})$, we directly do not know $\boldsymbol{u}$ while we can know that $\bar{\boldsymbol{u}} = 01001$ has an insertion error by majority decision with respect to lengths of substrings of $\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})$.

To find $\boldsymbol{u}$, we search $\sigma(\boldsymbol{u})$ and $\pi(\boldsymbol{u})$ which satisfy (14) and (15) for $\alpha = 1$, respectively. Note that we can know $\alpha = 1$ which is equivalent to the fact that $\boldsymbol{u}$ consists of two kinds of symbols because numbers of 0s and 1s in $\bar{\boldsymbol{u}}$ are greater than one.

For $\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})\backslash[\bar{\boldsymbol{u}}]$, there exist $g \in \Sigma$ and $h \in \Sigma$ such that $\sigma(\boldsymbol{u})g \in \overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})\backslash[\bar{\boldsymbol{u}}]$ and $h\pi(\boldsymbol{u}) \in \overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})\backslash[\bar{\boldsymbol{u}}]$ because $\tilde{\boldsymbol{x}}$ is a circular string. Therefore, for $\boldsymbol{s}' \in \overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})\backslash[\bar{\boldsymbol{u}}]$ and $c' \in \Sigma$, we check if $\pi(\boldsymbol{s}')$ and $c'\pi(\pi(\boldsymbol{s}'))$ satisfy (14) and (15) for $\alpha = 1$, respectively. Note that $\pi(\boldsymbol{s}')$ and $c'\pi(\pi(\boldsymbol{s}'))$ are candidates $\sigma(\boldsymbol{u})$ and $\pi(\boldsymbol{u})$, respectively.

Fig. 1 shows illustration of the relationship between $\pi(\boldsymbol{s}')$ and $c'\pi(\pi(\boldsymbol{s}'))$ for $\boldsymbol{u}$, where $\boldsymbol{u} = e\sigma(\boldsymbol{u}) = \pi(\boldsymbol{u})f$ and $e, f \in \Sigma$.

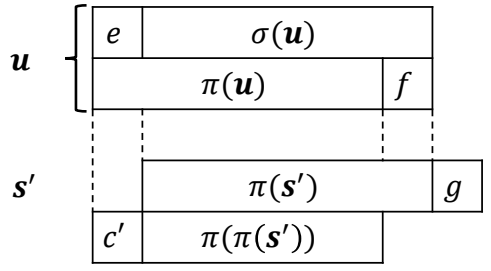


Fig. 1 Illustration of the relationship between $\pi(\boldsymbol{s}')$ and $c'\pi(\pi(\boldsymbol{s}'))$ for $\boldsymbol{u}$.

In this example, for $\boldsymbol{s}' = 1000 \in \overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})\backslash[01001]$ and $c' = 0$, we have that $\pi(\boldsymbol{s}') = 100$ and $c'\pi(\pi(\boldsymbol{s}')) = 010$ satisfy (14) and (15) for $\alpha = 1$, respectively. Indeed, we have that

$$\overline{C}_\sigma(100) = \overline{C}(1000) + \overline{C}(1001) - 1, \tag{24}$$

$$\overline{C}_\pi(010) = \overline{C}(0010) + \overline{C}(1010) - 1, \tag{25}$$

where $\alpha = 1$. Note that $\overline{C}_\sigma(100) = \overline{C}(1001) = 0, \overline{C}(1000) = 1$, and $\overline{C}_\pi(010) = 3, \overline{C}(0010) = 1, \overline{C}(1010) = 3$. From the left-hand sides of (14) and (15), $\sigma(\boldsymbol{u}) = 100$ and $\pi(\boldsymbol{u}) = 010$. Therefore, from (20), we can obtain $\boldsymbol{u} = 0100$. Hence, we can obtain $\mathcal{S}_4(\tilde{\boldsymbol{x}})$ in (7) from $\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})$ in (23) by replacing $\bar{\boldsymbol{u}} = 01001$ to $\boldsymbol{u} = 0100$.

4.2 Error Detection and Correction of Flipping Error

We first show Lemmas 2, 3, and 4 to find the position of flipping error in $\bar{\boldsymbol{u}}$. Note that each $\boldsymbol{v} \in \{\pi(\boldsymbol{u}), \sigma(\boldsymbol{u}), \pi(\bar{\boldsymbol{u}}), \sigma(\bar{\boldsymbol{u}})\}$ which is not shown in the equations of the lemmas satisfies (6).

Lemma 2: For $\boldsymbol{u}$ and $\bar{\boldsymbol{u}}$ such that

$$\pi(\boldsymbol{u}) = \pi(\bar{\boldsymbol{u}}) \; and \; \sigma(\boldsymbol{u}) \neq \sigma(\bar{\boldsymbol{u}}), \tag{26}$$

we have

$$\overline{C}_\pi(\sigma(\boldsymbol{u})) = \overline{C}(a\sigma(\boldsymbol{u})) + \overline{C}(\bar{a}\sigma(\boldsymbol{u})) + 1, \tag{27}$$

$$\overline{C}_\pi(\sigma(\bar{\boldsymbol{u}})) = \overline{C}(c\sigma(\bar{\boldsymbol{u}})) + \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) - 1, \tag{28}$$

where $a, c \in \Sigma$. Eqs. (27) and (28) also imply (29) and (30), respectively.

$$\overline{C}_\sigma(\sigma(\boldsymbol{u})) = \overline{C}(\sigma(\boldsymbol{u})b) + \overline{C}(\sigma(\boldsymbol{u})\bar{b}) - 1. \tag{29}$$

$$\overline{C}_\sigma(\sigma(\bar{\boldsymbol{u}})) = \overline{C}(\sigma(\bar{\boldsymbol{u}})d) + \overline{C}(\sigma(\bar{\boldsymbol{u}})\bar{d}) + 1. \tag{30}$$

Proof.

Proof of (27): Since $\sigma(\boldsymbol{u}) \neq \sigma(\bar{\boldsymbol{u}})$ from (26), $a\sigma(\boldsymbol{u}) \neq \bar{\boldsymbol{u}}$ and $\bar{a}\sigma(\boldsymbol{u}) \neq \bar{\boldsymbol{u}}$ hold. Moreover, since $\boldsymbol{u} = a\sigma(\boldsymbol{u})$ without loss of generality, we have

$$C(a\sigma(\boldsymbol{u})) + C(\bar{a}\sigma(\boldsymbol{u})) = \overline{C}(a\sigma(\boldsymbol{u})) + \overline{C}(\bar{a}\sigma(\boldsymbol{u})) + 1. \tag{31}$$

Moreover, from (26),

$$C_\pi(\sigma(\boldsymbol{u})) = \overline{C}_\pi(\sigma(\boldsymbol{u})). \tag{32}$$

From (3), (4), and (6),

$$C_\pi(\sigma(\boldsymbol{u})) = C(a\sigma(\boldsymbol{u})) + C(\bar{a}\sigma(\boldsymbol{u})). \tag{33}$$

From (31), (32), and (33), we have that (27) holds.

Proof of (28): From (26), $\sigma(\boldsymbol{u}) \neq \sigma(\bar{\boldsymbol{u}})$ holds, so that $\boldsymbol{u} \neq c\sigma(\bar{\boldsymbol{u}})$ and $\boldsymbol{u} \neq \bar{c}\sigma(\bar{\boldsymbol{u}})$ hold. Moreover, $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ without loss of generality, therefore,

$$C(c\sigma(\bar{\boldsymbol{u}})) + C(\bar{c}\sigma(\bar{\boldsymbol{u}})) = \overline{C}(c\sigma(\bar{\boldsymbol{u}})) + \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) - 1. \tag{34}$$

Moreover, from (26),

$$C_\pi(\sigma(\bar{\boldsymbol{u}})) = \overline{C}_\pi(\sigma(\bar{\boldsymbol{u}})). \tag{35}$$

From (3), (4), and (6),

$$C_\pi(\sigma(\bar{\boldsymbol{u}})) = C(c\sigma(\bar{\boldsymbol{u}})) + C(\bar{c}\sigma(\bar{\boldsymbol{u}})). \tag{36}$$

From (34), (35), and (36), we have that (28) holds.

As for (29) and (30), the right-hand side of (27) can be written by $\overline{C}_\sigma(\sigma(\boldsymbol{u})) + 1$ and the right-hand side of (28) can be written by $\overline{C}_\sigma(\sigma(\bar{\boldsymbol{u}})) - 1$. Therefore, (29) and (30) are satisfied. It completes the proof. □

Lemma 3: For $\boldsymbol{u}$ and $\bar{\boldsymbol{u}}$ such that only one bit is different between $\boldsymbol{u}$ and $\bar{\boldsymbol{u}}$,

$$\pi(\boldsymbol{u}) \neq \pi(\bar{\boldsymbol{u}}), \; and \; \sigma(\boldsymbol{u}) \neq \sigma(\bar{\boldsymbol{u}}), \tag{37}$$

we have

$$\overline{C}_\pi(\sigma(\boldsymbol{u})) = \overline{C}(a\sigma(\boldsymbol{u})) + \overline{C}(\bar{a}\sigma(\boldsymbol{u})) + \alpha, \tag{38}$$

$$\overline{C}_\sigma(\pi(\boldsymbol{u})) = \overline{C}(\pi(\boldsymbol{u})b) + \overline{C}(\pi(\boldsymbol{u})\bar{b}) + \beta, \tag{39}$$

$$\overline{C}_\pi(\sigma(\bar{\boldsymbol{u}})) = \overline{C}(c\sigma(\bar{\boldsymbol{u}})) + \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) - \gamma, \tag{40}$$

$$\overline{C}_\sigma(\pi(\bar{\boldsymbol{u}})) = \overline{C}(\pi(\bar{\boldsymbol{u}})d) + \overline{C}(\pi(\bar{\boldsymbol{u}})\bar{d}) - \delta, \tag{41}$$

where integers $0 \leq \alpha, \beta, \gamma, \delta \leq 2$ and $a, b, c, d \in \Sigma$.

Indeed, $\alpha = \beta = 0$ when $\boldsymbol{u} = e^k$ while $\alpha > 0$ and $\beta > 0$ when $\boldsymbol{u} \neq e^k$ ($e \in \Sigma$). Similarly, $\gamma = \delta = 0$ when $\bar{\boldsymbol{u}} = f^k$

while $\gamma > 0$ and $\delta > 0$ when $\bar{\boldsymbol{u}} \neq f^k$ ($f \in \Sigma$).

Proof. Proof of (38): From the proof of (27) in Lemma 2,

$$C(a\sigma(\boldsymbol{u})) + C(\bar{a}\sigma(\boldsymbol{u})) = \overline{C}(a\sigma(\boldsymbol{u})) + \overline{C}(\bar{a}\sigma(\boldsymbol{u})) + 1 \tag{42}$$

holds.

As for the left-hand side of (38), either

$$C_\pi(\sigma(\boldsymbol{u})) = \overline{C}_\pi(\sigma(\boldsymbol{u})) + 1, \text{ or} \tag{43}$$
$$C_\pi(\sigma(\boldsymbol{u})) = \overline{C}_\pi(\sigma(\boldsymbol{u})), \text{ or} \tag{44}$$
$$C_\pi(\sigma(\boldsymbol{u})) = \overline{C}_\pi(\sigma(\boldsymbol{u})) - 1 \tag{45}$$

holds. When $\boldsymbol{u} = \sigma(\boldsymbol{u})e$, that is, when $\boldsymbol{u} = e^k$ for $e \in \Sigma$, we have that (43) holds since $\pi(\boldsymbol{u}) = \sigma(\boldsymbol{u})$. Therefore, (44) and (45) hold for $\boldsymbol{u}$ having two kinds of letters. We also observe that (44) holds when $\pi(\bar{\boldsymbol{u}}) \neq \sigma(\boldsymbol{u})$, while (45) holds when $\pi(\bar{\boldsymbol{u}}) = \sigma(\boldsymbol{u})$.

From (3), (4), and (6),

$$C_\pi(\sigma(\boldsymbol{u})) = C(a\sigma(\boldsymbol{u})) + C(\bar{a}\sigma(\boldsymbol{u})). \tag{46}$$

Therefore, from (42), (43), and (46), we have that (38) holds when $\alpha = 0$. From (42), (44), and (46), (38) holds when $\alpha = 1$, while from (42), (45), and (46), (38) holds when $\alpha = 2$. Hence, (38) holds for any $\boldsymbol{u}$ satisfying (37).

Proof of (39): (39) also holds from the symmetry of $\tilde{\boldsymbol{x}}$ because (38) holds. Therefore, we have that $\alpha = \beta = 0$ when $\boldsymbol{u} = e^k$.

Proof of (40): From the proof of (34) in Lemma 2,

$$C(c\sigma(\bar{\boldsymbol{u}})) + C(\bar{c}\sigma(\bar{\boldsymbol{u}})) = \overline{C}(c\sigma(\bar{\boldsymbol{u}})) + \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) - 1. \tag{47}$$

As for the left-side hand of (40), either

$$C_\pi(\sigma(\bar{\boldsymbol{u}})) = \overline{C}_\pi(\sigma(\bar{\boldsymbol{u}})) + 1, \text{ or} \tag{48}$$
$$C_\pi(\sigma(\bar{\boldsymbol{u}})) = \overline{C}_\pi(\sigma(\bar{\boldsymbol{u}})), \text{ or} \tag{49}$$
$$C_\pi(\sigma(\bar{\boldsymbol{u}})) = \overline{C}_\pi(\sigma(\bar{\boldsymbol{u}})) - 1 \tag{50}$$

holds. From the proof of the left-hand side of (38), we have that (48), (49), and (50) also hold. Then, (48) holds when $\bar{\boldsymbol{u}} = f^k$ for $f \in \Sigma$ and $\gamma = 0$.

From (3), (4), and (6),

$$C_\pi(\sigma(\bar{\boldsymbol{u}})) = C(a\sigma(\bar{\boldsymbol{u}})) + C(\bar{a}\sigma(\bar{\boldsymbol{u}})). \tag{51}$$

From (47), (48), (49), (50), and (51), we have that (40) holds.

Proof of (41): (41) also holds from the symmetry of $\tilde{\boldsymbol{x}}$ because (40) holds. Therefore, we have that $\gamma = \delta = 0$ when $\bar{\boldsymbol{u}} = f^k$. It completes the proof. □

Lemma 4: For $\boldsymbol{u}$ and $\bar{\boldsymbol{u}}$ such that

$$\pi(\boldsymbol{u}) \neq \pi(\bar{\boldsymbol{u}}) \text{ and } \sigma(\boldsymbol{u}) = \sigma(\bar{\boldsymbol{u}}), \tag{52}$$

we have

$$\overline{C}_\sigma(\pi(\boldsymbol{u})) = \overline{C}(\pi(\boldsymbol{u})b) + \overline{C}(\pi(\boldsymbol{u})\bar{b}) + 1, \tag{53}$$

$$\overline{C}_\sigma(\pi(\bar{\boldsymbol{u}})) = \overline{C}(\pi(\bar{\boldsymbol{u}})d) + \overline{C}(\pi(\bar{\boldsymbol{u}})\bar{d}) - 1, \tag{54}$$

where $a, c \in \Sigma$. Eqs. (53) and (54) also imply (55) and (56), respectively.

$$\overline{C}_\pi(\pi(\boldsymbol{u})) = \overline{C}(a\pi(\boldsymbol{u})) + \overline{C}(\bar{a}\pi(\boldsymbol{u})) - 1. \tag{55}$$
$$\overline{C}_\pi(\pi(\bar{\boldsymbol{u}})) = \overline{C}(c\pi(\bar{\boldsymbol{u}})) + \overline{C}(\bar{c}\pi(\bar{\boldsymbol{u}})) + 1. \tag{56}$$

Proof. From Lemma 2 and the symmetry of $\tilde{\boldsymbol{x}}$, Lemma 4 is also satisfied. □

Note that $a'' \in \Sigma$ and $b'' \in \Sigma$ exist such that $\sigma(\boldsymbol{u})a'' \in \overline{S}_k(\tilde{\boldsymbol{x}})$ and $b''\pi(\boldsymbol{u}) \in \overline{S}_k(\tilde{\boldsymbol{x}})$, respectively.

Lemmas 2, 3, and 4 can detect a string in $\overline{S}_k(\tilde{\boldsymbol{x}})$ such that a bit-flipping error is occurred in the last symbol of $\bar{\boldsymbol{u}}$, in a symbol of $\bar{\boldsymbol{u}}$, except for the top and last symbols, in the top symbol of $\bar{\boldsymbol{u}}$, respectively.

We show Propositions 2, 3, and 4 to determine $\bar{\boldsymbol{u}}$ in $\overline{S}_k(\tilde{\boldsymbol{x}})$ and to correct $\bar{\boldsymbol{u}}$ to $\boldsymbol{u}$ for the cases that Lemmas 2, 3, and 4 are satisfied, respectively.

To prove Proposition 2, we first prove Lemmas 5 and 6.

Lemma 5: For $\boldsymbol{w} \in \Sigma^*$ and $a, b \in \Sigma$, if $a\boldsymbol{w}b, a\boldsymbol{w}\bar{b}, \bar{a}\boldsymbol{w}b \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$, then $\bar{a}\boldsymbol{w}\bar{b} \notin \mathcal{S}_k(\tilde{\boldsymbol{x}})$.

Proof. If $\bar{a}\boldsymbol{w}\bar{b} \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$, then there are two pairs of exceptional substrings because $a\boldsymbol{w}b$ and $a\boldsymbol{w}\bar{b}$ are one pair of exceptional substrings while $\bar{a}\boldsymbol{w}b$ and $\bar{a}\boldsymbol{w}\bar{b}$ are also the other pair of exceptional substrings. It contradicts Assumption 2. Therefore, $\bar{a}\boldsymbol{w}\bar{b} \notin \mathcal{S}_k(\tilde{\boldsymbol{x}})$. □

Lemma 6: For $p\boldsymbol{v}, \boldsymbol{v}q \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ where $p, q \in \Sigma$ and $\boldsymbol{v} \in \Sigma^*$, if $C(p\boldsymbol{v}) = C(\boldsymbol{v}q) = 1$ and $C_\pi(\boldsymbol{v}) = 1$ are satisfied, then $p\boldsymbol{v}q$ of length $k+1$ is a substring of $\tilde{\boldsymbol{x}}$.

Proof. Since $C_\pi(\boldsymbol{v}) = 1$, $\boldsymbol{v}$ is the unique substring of $\tilde{\boldsymbol{x}}$. Moreover, $C(p\boldsymbol{v}) = C(\boldsymbol{v}q) = 1$ holds, so that $p\boldsymbol{v}q$ is a substring of $\tilde{\boldsymbol{x}}$. □

Remark 1: From Assumption 2, we can reconstruct $\tilde{\boldsymbol{x}}$ by using $\mathcal{S}_k(\tilde{\boldsymbol{x}})$. Therefore, there exists $\boldsymbol{s} \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ such that $\sigma(\boldsymbol{s}) = \pi(p\boldsymbol{v})$. In other words, $\boldsymbol{s}$ overlaps with $p\boldsymbol{v}$. When repeating the overlapping manner from $p\boldsymbol{v}, \boldsymbol{v}q$ is the last one which is overlapped because $p\boldsymbol{v}q$ is the unique substring of $\tilde{\boldsymbol{x}}$ from Lemma 6.

The proof of Proposition 2 is in Appendix A.

Proposition 2: For $\overline{S}_k(\tilde{\boldsymbol{x}})$ such that $\bar{\boldsymbol{u}}$ having one bit-flipping error satisfies (26), we can determine $\bar{\boldsymbol{u}}$ and correct $\bar{\boldsymbol{u}}$ to $\boldsymbol{u}$.

Proof. See Appendix A.

Proposition 3: For $\overline{S}_k(\tilde{\boldsymbol{x}})$ such that $\bar{\boldsymbol{u}}$ has one bit-flipping error and satisfies (37), we can determine $\bar{\boldsymbol{u}}$ and correct $\bar{\boldsymbol{u}}$ to $\boldsymbol{u}$.

Proof. We first prove the case such that both $\boldsymbol{u}$ and $\bar{\boldsymbol{u}}$ consist of two kinds of letters in Σ.

From (38), (39), (40), and (41), $\sigma(\boldsymbol{u})$, $\pi(\boldsymbol{u})$, $\sigma(\bar{\boldsymbol{u}})$,

and $\pi(\bar{\boldsymbol{u}})$ are obtained, respectively, because $\alpha \neq 0$, $\beta \neq 0$, $\gamma \neq 0$, and $\delta \neq 0$ from Lemma 3. Therefore, for the first symbol a of $\pi(\boldsymbol{u})$ and the last symbol b of $\sigma(\boldsymbol{u})$, we can obtain

$$\boldsymbol{u} = a\pi(\sigma(\boldsymbol{u}))b, \tag{57}$$

$$\bar{\boldsymbol{u}} = a\pi(\sigma(\bar{\boldsymbol{u}}))b, \tag{58}$$

because (37) holds.

We next prove the case such that either $\boldsymbol{u}$ or $\bar{\boldsymbol{u}}$ is a string consisting of a single letter because $\boldsymbol{u} \neq \bar{\boldsymbol{u}}$.

We consider that $\boldsymbol{u}$ is a string consisting of a single letter. In this case, from Lemma 3, $\alpha = \beta = 0$ while $\gamma > 0$ and $\delta > 0$. Therefore, $\pi(\bar{\boldsymbol{u}})$ and $\sigma(\bar{\boldsymbol{u}})$ can be obtained, so that for the first symbol c of $\pi(\bar{\boldsymbol{u}})$ and the last symbol d of $\sigma(\bar{\boldsymbol{u}})$, we have

$$\bar{\boldsymbol{u}} = c\pi(\sigma(\bar{\boldsymbol{u}}))d. \tag{59}$$

From (37) and $n \geq 3$, $c = d$ holds, and therefore,

$$\boldsymbol{u} = c^k. \tag{60}$$

Since $\bar{\boldsymbol{u}}$ is a string consisting of a single letter, from the proof of the case such that $\bar{\boldsymbol{u}}$ is c^k, we have

$$\boldsymbol{u} = e\pi(\sigma(\boldsymbol{u}))f, \tag{61}$$

$$\bar{\boldsymbol{u}} = e^k, \tag{62}$$

where the first symbol e of $\pi(\boldsymbol{u})$ and the last symbol f of $\sigma(\boldsymbol{u})$. It completes the proof. □

Proposition 4: For $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ such that $\bar{\boldsymbol{u}}$ has one bit-flipping error and satisfies (52), we can determine $\bar{\boldsymbol{u}}$ and correct $\bar{\boldsymbol{u}}$ to $\boldsymbol{u}$.

Proof. From Proposition 2 and the symmetry of $\tilde{\boldsymbol{x}}$, Proposition 4 is also satisfied. □

We will show two examples for a flipping error. For $\boldsymbol{x} = 0001011011$ and $k = 4$,

$$\mathcal{S}_4(\tilde{\boldsymbol{x}}) = [0001, 0010, 0101, 0110, 0110, 1000, 1011, 1011, 1100, 1101]. \tag{63}$$

First, for example, consider the case when one bit-flipping error occurs at the last symbol of 1011 in (63); that is

$$\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}}) = [0001, 0010, 0101, 0110, 0110, 1000, 1011, \mathbf{1010}, 1100, 1101]. \tag{64}$$

The error detection and correction scheme of the flipping error are as follows. For given $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$, we need to find strings of length k which satisfy any of the Lemmas 2, 3, or 4 because $\boldsymbol{u}$ and $\bar{\boldsymbol{u}}$ are unknown.

As for examples of flipping errors, we will use a brute-force search to find the strings satisfying the lemmas to simplify explanations. Note that we can use substrings of strings in $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ as candidates of the strings satisfying the lemmas to reduce computational time as shown in the example of insertion error in Sec. 4.1.

By applying brute-force search for strings of length 4, for $a \in \Sigma$, we obtain strings $a011$ and $\bar{a}011$ which satisfy (27) in Lemma 2. Similarly, for $b \in \Sigma$, we obtain strings $b010$ and $\bar{b}010$ which satisfy (28) in Lemma 2. Therefore, Lemma 2 is satisfied in this case. Indeed, we have that

$$\overline{C}_\pi(011) = \overline{C}(0011) + \overline{C}(1011) + 1, \tag{65}$$

$$\overline{C}_\pi(010) = \overline{C}(0010) + \overline{C}(1010) - 1, \tag{66}$$

hold. Note that $\overline{C}_\pi(011) = 2, \overline{C}(0011) = 0$, and $\overline{C}(1011) = 1$ Then, $\overline{C}_\pi(010) = 1$ and $\overline{C}(0010) = \overline{C}(1010) = 1$.

Next, we determine $\bar{\boldsymbol{u}}$ from the case in proof of Proposition 2 by using $\sigma(\boldsymbol{u}) = 011$ and $\sigma(\bar{\boldsymbol{u}}) = 010$. In this case, (A· 15) and (A· 8) are satisfied in the proof of Proposition 2. Since $\overline{C}(0011) = 0$ and $\overline{C}(0010) = 1$, we have $\overline{C}_\pi(001) = 1$. Moreover, from $C_\pi(001) = 1$, there exists symbol $a(= 0)$ such that $\overline{C}(a001) = \overline{C}(0001) = 1$. From Assumption 2, we can reconstruct $\tilde{\boldsymbol{x}}$ by using $\mathcal{S}_4(\tilde{\boldsymbol{x}})$. To determine $d \in \Sigma$ such that $0001d$ is a substring of $\tilde{\boldsymbol{x}}$ by using Lemma 6, we repeat the overlapping manner from 0001 for $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ until $1011(=c\sigma(\boldsymbol{u}))$ or $1010(=c\sigma(\bar{\boldsymbol{u}}))$ appears.

$$0001 \rightarrow 1000 \rightarrow 1100 \rightarrow 0110 \rightarrow 1011 \tag{67}$$

From (67), 1011 appears before 1010, so that $\bar{\boldsymbol{u}} = 1010$ and $d = 0$ from Proposition 2. Therefore, $\boldsymbol{u} = 1011$ and 00010 is a substring of $\tilde{\boldsymbol{x}}$. We obtain $\mathcal{S}_4(\tilde{\boldsymbol{x}})$ shown in (63) from $\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})$ in (64) by replacing $\bar{\boldsymbol{u}} = 1010$ to $\boldsymbol{u} = 1011$.

Second, for example, consider the case when one bit-flipping error occurs at the third symbol of 0010 in (63); that is

$$\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}}) = [0001, \mathbf{0000}, 0100, 0101, 0101, 0101, 1000, 1010, 1010, 1010]. \tag{68}$$

By applying brute-force search for strings of length 4, for $a \in \Sigma$, we obtain strings $a010$ and $\bar{a}010$ which satisfy (38) for $\alpha > 0$ in Lemma 3. Similarly, for $b \in \Sigma$, we obtain $001b$ and $001\bar{b}$ which satisfy (39) for $\beta > 0$ in Lemma 3. Therefore, Lemma 3 is satisfied in this case. From (38) and (39), we obtain $\sigma(\boldsymbol{u}) = 010$ and $\pi(\boldsymbol{u}) = 001$.

On the other hand, we cannot find strings which satisfy (40) for $\gamma > 0$ and (41) for $\delta > 0$. It follows that we have $\gamma = \delta = 0$ for (40) and (41) because we find the strings satisfying (38) for $\alpha > 0$ and (39) for $\beta > 0$. Since both of the first symbol of $\pi(\boldsymbol{u})$ and the last symbol of $\sigma(\boldsymbol{u})$ are 0, we obtain $\bar{\boldsymbol{u}} = 0000$. Indeed, we have that

$$\overline{C}_\pi(010) = \overline{C}(0010) + \overline{C}(1010) + 1(= \alpha), \tag{69}$$

$$\overline{C}_\sigma(001) = \overline{C}(0010) + \overline{C}(0011) + 1(= \beta), \tag{70}$$

$$\overline{C}_\pi(000) = \overline{C}(0000) + \overline{C}(1000) - 0(= \gamma), \tag{71}$$

$$\overline{C}_\sigma(000) = \overline{C}(0000) + \overline{C}(0001) - 0(= \delta), \tag{72}$$

hold. Note that $\overline{C}_\pi(010) = 4, \overline{C}(0010) = 0, \overline{C}(1010) = 3$, and $\overline{C}_\sigma(001) = 1, \overline{C}(0010) = \overline{C}(0011) = 0$. Then,

$\overline{C}_\pi(000) = 2, \overline{C}(0000) = \overline{C}(1000) = 1$, and $\overline{C}_\sigma(000) = 2, \overline{C}(0000) = \overline{C}(0001) = 1$. Since $\pi(\boldsymbol{u}) = 001, \sigma(\boldsymbol{u}) = 010$, $\pi(\bar{\boldsymbol{u}}) = \sigma(\bar{\boldsymbol{u}}) = 000$, we have $\boldsymbol{u} = 0010$ and $\bar{\boldsymbol{u}} = 0000$ from Lemma 3 and Proposition 3. We can obtain $\mathcal{S}_4(\tilde{\boldsymbol{x}})$ shown in (63) from $\overline{\mathcal{S}}_4(\tilde{\boldsymbol{x}})$ in (68) by replacing $\bar{\boldsymbol{u}} = 0000$ to $\boldsymbol{u} = 0010$.

In a noiseless situation, the proposed algorithm can be used to identify the substring such that the first symbol is x_1 by embedding one bit-flipping (resp. insertion and deletion) error to the substring when the substring is the unique substring of length k in $\tilde{\boldsymbol{x}}$. This technique is useful for retrieving $\boldsymbol{x}$ from $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ and the reconstructed $\tilde{\boldsymbol{x}}$. For example, the technique can remove one of the codewords in CSE, called rank($\boldsymbol{x}$) [5], which is used to identify the position of x_1 in $\tilde{\boldsymbol{x}}$.

5. Conclusion

In this paper, we proposed an error correcting algorithm which can detect and correct a substring having one bit-flipping, one bit-insertion, or one bit-deletion error. By applying the proposed algorithm, a reconstruction algorithm [9] can assemble the input circular string from its substrings and their frequencies including only one substring which has at most one bit error.

In future works, we will extend the proposed algorithm to that over a finite alphabet, in order to apply the extended algorithm to DNA strands in DNA storage systems.

Acknowledgment

The authors would like to thank Prof. Hiroyoshi Morita and Dr. Hirotada Fukae for their helpful discussions. The authors are grateful to anonymous reviewers ' valuable comments that improved the manuscript. This work was supported by JSPS KAKENHI Grant Number JP23K10983.

References

[1] T. Ota and A. Manada, "One bit-flipping/insertion/deletion correcting code for substrings of binary circular string," Proc. of the International Symposium on Information Theory and its Applications, pp.159–164, 2024.
[2] D. Dubé and V. Beaudoin, "Lossless data compression via substring enumeration," Proc. of the Data Compression Conference, pp.229–238, Mar. 2010.
[3] O. Sabary, A. Yucovich, G. Shapira, and E. Yaakobi, "Reconstruction algorithms for DNA-storage systems," Scientific Reports, vol.14, no.1951, 2024.
[4] M. Béliveau and D. Dubé, "Improving compression via substring enumeration by explicit phase awareness," Proc. of the Data Compression Conference 2014, pp.26–28, Mar. 2014.
[5] S. Kanai, H. Yokoo, K. Yamazaki, and H. Kaneyasu, "Efficient implementation and empirical evaluation of compression by substring enumeration," IEICE Trans. on Fundamentals, vol.E99-A, no.2, pp.601–611, 2016.
[6] T. Ota and H. Morita, "A compact tree representation of an antidictionary," IEICE Trans. on Fundamentals, vol.E100-A, no.9, pp.587–591, 2017.
[7] T. Ota, H. Morita, and A. Manada, "Compression by substring enumeration using sorted contingency tables," IEICE Transactions on Fundamentals, vol.E103-A, no.6, pp.829–835, 2020.
[8] "bzip2." `https://www.sourceware.org/bzip2/`.
[9] T. Ota and A. Manada, "A reconstruction of circular binary string using substrings and minimal absent words," IEICE Transactions on Fundamentals, vol.E107-A, no.3, pp.409–416, 2024.
[10] S. Yazdi, H. Kiah, E. Garcia, J. Ma, H. Zhao, and O. Milenkovic, "DNA-based storage: Trends and methods," IEEE Trans. Molecular, Biological, Multi-Scale Commun., vol.1, no.3, pp.230–248, 2015.

Appendix A: Proof of Proposition 2

We will first focus on how to detect $\bar{\boldsymbol{u}}$ from $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$ because $\boldsymbol{u}$ can be obtained by flipping the last symbol of $\bar{\boldsymbol{u}}$.

Candidates of $\bar{\boldsymbol{u}}$ are $c\sigma(\bar{\boldsymbol{u}})$ and $\bar{c}\sigma(\bar{\boldsymbol{u}})$ for $c \in \Sigma$ because we know $\sigma(\bar{\boldsymbol{u}})$ from (28). To determine $\bar{\boldsymbol{u}}$, we consider two cases with respect to $c\sigma(\bar{\boldsymbol{u}})$ and $\bar{c}\sigma(\bar{\boldsymbol{u}})$ as follows.

(a) $\overline{C}(c\sigma(\bar{\boldsymbol{u}})) \geq 2$ or $\overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) \geq 2$
(b) $\overline{C}(c\sigma(\bar{\boldsymbol{u}})) \leq 1$ and $\overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) \leq 1$

We first show case (a). Suppose that $\overline{C}(c\sigma(\bar{\boldsymbol{u}})) = \ell \geq 2$ without loss of generality. As for $\overline{C}(c\sigma(\bar{\boldsymbol{u}}))$ and $\overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}}))$, there are five disjoint cases:

$$\overline{C}(c\sigma(\bar{\boldsymbol{u}})) = \ell \text{ and } \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 0, \text{or} \quad (\text{A}\cdot 1)$$
$$\overline{C}(c\sigma(\bar{\boldsymbol{u}})) = \ell \text{ and } \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 1, \text{or} \quad (\text{A}\cdot 2)$$
$$\overline{C}(c\sigma(\bar{\boldsymbol{u}})) = \ell \text{ and } 2 \leq \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) < \ell, \text{or} \quad (\text{A}\cdot 3)$$
$$\overline{C}(c\sigma(\bar{\boldsymbol{u}})) = \ell \text{ and } \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) = \ell, \text{or} \quad (\text{A}\cdot 4)$$
$$\overline{C}(c\sigma(\bar{\boldsymbol{u}})) = \ell \text{ and } \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) > \ell. \quad (\text{A}\cdot 5)$$

When (A· 1) holds, we obtain $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ because $\overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 0$.

When (A· 2) holds, any of the following four relations with respect to $\sigma(\boldsymbol{u})$

$$\overline{C}(c\sigma(\boldsymbol{u})) > 0 \text{ and } \overline{C}(\bar{c}\sigma(\boldsymbol{u})) > 0, \text{or} \quad (\text{A}\cdot 6)$$
$$\overline{C}(c\sigma(\boldsymbol{u})) > 0 \text{ and } \overline{C}(\bar{c}\sigma(\boldsymbol{u})) = 0, \text{or} \quad (\text{A}\cdot 7)$$
$$\overline{C}(c\sigma(\boldsymbol{u})) = 0 \text{ and } \overline{C}(\bar{c}\sigma(\boldsymbol{u})) > 0, \text{or} \quad (\text{A}\cdot 8)$$
$$\overline{C}(c\sigma(\boldsymbol{u})) = 0 \text{ and } \overline{C}(\bar{c}\sigma(\boldsymbol{u})) = 0 \quad (\text{A}\cdot 9)$$

holds.

When (A· 2) and (A· 6) hold, from Lemma 5, we obtain $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$.

When (A· 2) and (A· 7) hold, we have $\overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 1$ and $\overline{C}(\bar{c}\sigma(\boldsymbol{u})) = 0$. Therefore, $\overline{C}_\pi(\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ holds from (4) and (26). Moreover, since $\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})) = \pi(\bar{c}\sigma(\boldsymbol{u}))$, $C_\pi(\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ holds. Furthermore, there exists $a \in \Sigma$ such that $\overline{C}(a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$. Indeed, if $\overline{C}(a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) \geq 2$, then $\overline{C}(\bar{c}\sigma(\boldsymbol{u})) > 0$, however, it contradicts $\overline{C}(\bar{c}\sigma(\boldsymbol{u})) = 0$ in (A· 7). Hence, $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ because $C_\pi(\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ and the flipping error occurs in the last symbol.

We can use Lemma 6 by replacing $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))$ to $p\boldsymbol{v}$ and $\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))d$ to $\boldsymbol{v}q$ for some $d \in \Sigma$. From Lemma 6, $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))d$ is a substring of $\tilde{\boldsymbol{x}}$.

From Assumption 2, we can reconstruct $\tilde{\boldsymbol{x}}$ by using

$\mathcal{S}_k(\tilde{\boldsymbol{x}})$. Therefore, there exists $\boldsymbol{s} \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ such that $\sigma(\boldsymbol{s}) = \pi(a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})))$. In other words, $\boldsymbol{s}$ overlaps with $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))$. We now determine d by using Lemma 6. Suppose that we repeat the overlapping manner from $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))$ for $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$. If $c\sigma(\boldsymbol{u})$ appears before $c\sigma(\bar{\boldsymbol{u}})$, then $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\bar{\boldsymbol{u}})$ from Lemma 6. In this case, $\bar{c}\sigma(\bar{\boldsymbol{u}})$ is the last substring $\boldsymbol{v}q$ in Remark 1. On the other hand, if $c\sigma(\bar{\boldsymbol{u}})$ appears before $c\sigma(\boldsymbol{u})$, then $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\boldsymbol{u})$ from Lemma 6 In this case, $\bar{c}\sigma(\boldsymbol{u})$ is the last substring $\boldsymbol{v}q$ in Remark 1.

When (A· 2) and (A· 8) hold, from Lemma 5, we obtain $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$.

When (A· 2) and (A· 9) hold, the proof is similar to the case of (A· 2) and (A· 7). Since $\overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 1$ and $\overline{C}(\bar{c}\sigma(\boldsymbol{u})) = 0$, $\overline{C}_\pi(\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ holds from (4) and (26). Moreover, since $\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})) = \pi(\bar{c}\sigma(\boldsymbol{u}))$, $C_\pi(\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ holds. Furthermore, there exists $a \in \Sigma$ such that $\overline{C}(a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$.

We can use Lemma 6 by replacing $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))$ to $p\boldsymbol{v}$ and $\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))d$ to $\boldsymbol{v}q$ for some $d \in \Sigma$. From Lemma 6, $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))d$ is a substring of $\tilde{\boldsymbol{x}}$. Moreover, from (A· 9) and the fact that there exists only one substring $\bar{\boldsymbol{u}}$ having a flipping error, $C(\boldsymbol{u}) = 1$ and $C_\pi(\sigma(\boldsymbol{u})) = 1$. Since $C_\pi(\sigma(\boldsymbol{u})) = 1$, there exists $e \in \Sigma$ such that $C(\sigma(\boldsymbol{u})e) = 1$ and $C(\sigma(\boldsymbol{u})\bar{e}) = 0$. Therefore, from Lemma 6, $\boldsymbol{u}e$ is a substring of $\tilde{\boldsymbol{x}}$. Hence, $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))de$ is a substring of $\tilde{\boldsymbol{x}}$.

From Assumption 2, we can reconstruct $\tilde{\boldsymbol{x}}$ by using $\mathcal{S}_k(\tilde{\boldsymbol{x}})$. Therefore, there exists $\boldsymbol{s} \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ such that $\sigma(\boldsymbol{s}) = \pi(a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})))$. In other words, $\boldsymbol{s}$ can be overlapping to $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))$. To determine d by using Lemma 6, when we repeat the overlapping manner from $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))$ for $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$, if $\sigma(\boldsymbol{u})f$ appears before $\sigma(\bar{\boldsymbol{u}})g$, then $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\bar{\boldsymbol{u}})$ from Lemma 6, where $f, g \in \Sigma$. In this case, $\bar{c}\sigma(\bar{\boldsymbol{u}})$ is the last substring $\boldsymbol{v}q$ in Remark 1. Otherwise, that is if $\sigma(\bar{\boldsymbol{u}})g$ appears before $\sigma(\boldsymbol{u})f$, then $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\boldsymbol{u})$ from Lemma 6. In this case, $\bar{c}\sigma(\boldsymbol{u})$ is the last substring $\boldsymbol{v}q$ in Remark 1.

When (A· 3) holds, any of the four relations (A· 6), (A· 7), (A· 8), or (A· 9) holds.

From Lemma 5, (A· 3) and (A· 6) never hold.

When (A· 3) and (A· 7) hold, from Lemma 5, $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$.

When (A· 3) and (A· 8) hold, from Lemma 5, $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$.

When (A· 3) and (A· 9) hold, $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ because $\overline{C}(c\sigma(\bar{\boldsymbol{u}})) > \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}}))$ and there exists at most one pair of exceptional substrings from Assumption 2.

When (A· 4) holds, any of the four relations (A· 6), (A· 7), (A· 8), or (A· 9) holds. From condition (b) of Assumption 2, we only need to consider the case $\ell = 2$ because $c\sigma(\bar{\boldsymbol{u}})$ and $\bar{c}\sigma(\bar{\boldsymbol{u}})$ form a pair of exceptional substrings by reading these strings from right to left. Therefore, either $C(c\sigma(\bar{\boldsymbol{u}})) = 1$ or $C(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 1$ holds.

From Lemma 5, (A· 4) and (A· 6) never hold.

When (A· 4) and (A· 7) hold, from Lemma 5, $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$.

When (A· 4) and (A· 8) hold, from Lemma 5, $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$.

As for (A· 4) and (A· 9), we first consider that $c\sigma(\bar{\boldsymbol{u}})$ or $\bar{c}\sigma(\bar{\boldsymbol{u}})$ is a string consisting of a single letter. If $c\sigma(\bar{\boldsymbol{u}})$ is a string consisting of a single letter, then $c\sigma(\boldsymbol{u}) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ holds. Therefore, $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$. On the other hand, if $\bar{c}\sigma(\bar{\boldsymbol{u}})$ is a string consisting of a single letter, then $\bar{c}\sigma(\boldsymbol{u}) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ holds. Therefore, $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$.

We next consider that $c\sigma(\bar{\boldsymbol{u}})$ and $\bar{c}\sigma(\bar{\boldsymbol{u}})$ are strings including two kinds of letters. From Assumption 2, we can reconstruct $\tilde{\boldsymbol{x}}$ by using $\mathcal{S}_k(\tilde{\boldsymbol{x}})$. Therefore, there exist $a, b \in \Sigma$ such that $\overline{C}(a\pi(c\sigma(\bar{\boldsymbol{u}})) > 0$ and $\overline{C}(b\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})) > 0$. Moreover, $a\pi(c\sigma(\bar{\boldsymbol{u}})) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ and $b\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ are satisfied. Since $c\sigma(\bar{\boldsymbol{u}})$ is a string consisting of two kinds of letters, $\pi(c\sigma(\bar{\boldsymbol{u}})) \neq \sigma(\bar{\boldsymbol{u}})$. Hence,

$$a\pi(c\sigma(\bar{\boldsymbol{u}})) \neq c\sigma(\bar{\boldsymbol{u}}) \quad \text{(A· 10)}$$

$$a\pi(c\sigma(\bar{\boldsymbol{u}})) \neq \bar{c}\sigma(\bar{\boldsymbol{u}}) \quad \text{(A· 11)}$$

hold.

Similarly, since $\bar{c}\sigma(\bar{\boldsymbol{u}})$ is a string consisting of two kinds of letters, $\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})) \neq \sigma(\bar{\boldsymbol{u}})$. Hence,

$$b\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})) \neq c\sigma(\bar{\boldsymbol{u}}) \quad \text{(A· 12)}$$

$$b\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})) \neq \bar{c}\sigma(\bar{\boldsymbol{u}}) \quad \text{(A· 13)}$$

hold.

From (A· 4), $c\sigma(\bar{\boldsymbol{u}})$ and $\bar{c}\sigma(\bar{\boldsymbol{u}})$ form pair of exceptional substrings by reading the substrings from right to left.Therefore, from Assumption 2, (A· 10), (A· 11), (A· 12), and (A· 13), $\bar{a}\pi(c\sigma(\bar{\boldsymbol{u}})) \notin \mathcal{S}_k(\tilde{\boldsymbol{x}})$ and $\bar{b}\pi(c\sigma(\bar{\boldsymbol{u}})) \notin \mathcal{S}_k(\tilde{\boldsymbol{x}})$ are satisfied because there exists at most one pair of exceptional substrings. Since $c\sigma(\bar{\boldsymbol{u}})$ and $\bar{c}\sigma(\bar{\boldsymbol{u}})$ form a pair of exceptional substrings, from condition (b) of Assumption 2, $C(c\sigma(\bar{\boldsymbol{u}})) = 1$ or $C(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 1$ holds. Therefore, either $\overline{C}(a\pi(c\sigma(\bar{\boldsymbol{u}}))) = 1$ or $\overline{C}(b\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ holds. Hence, if $\overline{C}(a\pi(c\sigma(\bar{\boldsymbol{u}}))) = 1$, then $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$. On the other hand, if $\overline{C}(b\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$, then $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$.

When (A· 5) holds, any of the four relations (A· 6), (A· 7), (A· 8), or (A· 9) holds.

From Lemma 5, (A· 5) and (A· 6) never hold.

As for the other cases, from condition (b) of Assumption 2, we consider only $\ell = 2$ because $c\sigma(\bar{\boldsymbol{u}})$ and $\bar{c}\sigma(\bar{\boldsymbol{u}})$ are form a pair of exceptional substrings by reading the substrings from right to left. Therefore, $C(c\sigma(\bar{\boldsymbol{u}})) = 1$ or $C(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 1$ holds. Moreover, since $\overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) > \ell$ from (A· 5), we have $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ when (A· 5) and any of the three relations (A· 7), (A· 8), or (A· 9) hold.

Since all cases of (a) are proved, we can obtain $\bar{\boldsymbol{u}}$. Therefore, $\boldsymbol{u}$ can be obtained by flipping the last symbol of $\bar{\boldsymbol{u}}$. Hence, it completes the proof of case (a).

Next, we consider case (b). We suppose that $\overline{C}(c\sigma(\bar{\boldsymbol{u}})) = 1$ without loss of generality. Note that $\overline{C}(c\sigma(\bar{\boldsymbol{u}})) = 0$ and $\overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 0$ never hold because either $c\sigma(\bar{\boldsymbol{u}})$ or $\bar{c}\sigma(\bar{\boldsymbol{u}})$ has an flipping error. In this case, one of the following equations holds.

$$\overline{C}(c\sigma(\bar{\boldsymbol{u}})) = 1 \text{ and } \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 0, \text{or} \quad \text{(A· 14)}$$

$$\overline{C}(c\sigma(\bar{\boldsymbol{u}})) = 1 \text{ and } \overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 1. \quad (\text{A}\cdot 15)$$

When (A· 14) holds, $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ because $\overline{C}(\bar{c}\sigma(\bar{\boldsymbol{u}})) = 0$.

When (A· 15) holds, either $c\sigma(\bar{\boldsymbol{u}}) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ or $\bar{c}\sigma(\bar{\boldsymbol{u}}) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ holds, therefore, $\sigma(\bar{\boldsymbol{u}})$ is a substrings of $\tilde{\boldsymbol{x}}$. Moreover, any of the four relations (A· 6), (A· 7), (A· 8), or (A· 9) holds.

When (A· 15) and (A· 6) hold, we consider the following three disjoint cases:

$$\overline{C}(c\sigma(\boldsymbol{u})) > \overline{C}(\bar{c}\sigma(\boldsymbol{u})), \quad (\text{A}\cdot 16)$$

$$\overline{C}(c\sigma(\boldsymbol{u})) < \overline{C}(\bar{c}\sigma(\boldsymbol{u})), \quad (\text{A}\cdot 17)$$

$$\overline{C}(c\sigma(\boldsymbol{u})) = \overline{C}(\bar{c}\sigma(\boldsymbol{u})). \quad (\text{A}\cdot 18)$$

When (A· 15), (A· 6), and (A· 16) hold, we have $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$. To prove this, we use proof by contradiction. Assume, for the sake of contradiction, that $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$.

From (A· 6), $c\sigma(\boldsymbol{u})$ and $\bar{c}\sigma(\boldsymbol{u})$ form a pair of exceptional substrings by reading right to left. From (A· 6) and (A· 16), $\overline{C}(c\sigma(\boldsymbol{u})) \geq 2$ and $\overline{C}(\bar{c}\sigma(\boldsymbol{u})) \geq 1$ hold. Moreover, since $\overline{C}(\bar{c}\sigma(\boldsymbol{u})) \geq 1$ and $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$ from the assumption, we have $C(\bar{c}\sigma(\boldsymbol{u})) \geq 2$. However, it contradicts (b) of Assumption 2 because $C(c\sigma(\boldsymbol{u})) \geq 2$ and $C(\bar{c}\sigma(\boldsymbol{u})) \geq 2$. Hence, $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$.

By replacing $c\sigma(\boldsymbol{u})$ to $\bar{c}\sigma(\boldsymbol{u})$ in the proof that (A· 15), (A· 6), and (A· 16) hold, when (A· 15), (A· 6), and (A· 17) hold, $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$.

When (A· 15), (A· 6), and (A· 18) hold, we use the same manner as in the proof when (A· 4) and (A· 9) hold.

We first consider that $c\sigma(\boldsymbol{u})$ or $\bar{c}\sigma(\boldsymbol{u})$ is a string consisting of a single letter. If $c\sigma(\boldsymbol{u})$ is a string consisting of a single letter, then $c\sigma(\bar{\boldsymbol{u}}) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ holds. Therefore, $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$. On the other hand, if $\bar{c}\sigma(\boldsymbol{u})$ is a string consisting of a single letter, then $\bar{c}\sigma(\bar{\boldsymbol{u}}) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ holds. Therefore, $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$.

We next consider that $c\sigma(\boldsymbol{u})$ and $\bar{c}\sigma(\boldsymbol{u})$ are strings consisting of two kinds of letters. From Assumption 2, we can reconstruct $\tilde{\boldsymbol{x}}$ by using $\mathcal{S}_k(\tilde{\boldsymbol{x}})$. Therefore, there exist $a, b \in \Sigma$ such that $\overline{C}(a\pi(c\sigma(\boldsymbol{u}))) > 0$ and $\overline{C}(b\pi(\bar{c}\sigma(\boldsymbol{u}))) > 0$. Moreover, $a\pi(c\sigma(\boldsymbol{u})) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ and $b\pi(\bar{c}\sigma(\boldsymbol{u})) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ are satisfied. Since $c\sigma(\boldsymbol{u})$ is a string consisting of two kinds of letters, $\pi(c\sigma(\boldsymbol{u})) \neq \sigma(\boldsymbol{u})$. Hence,

$$a\pi(c\sigma(\boldsymbol{u})) \neq c\sigma(\boldsymbol{u}) \quad (\text{A}\cdot 19)$$

$$a\pi(c\sigma(\boldsymbol{u})) \neq \bar{c}\sigma(\boldsymbol{u}) \quad (\text{A}\cdot 20)$$

hold.

Similarly, since $\bar{c}\sigma(\boldsymbol{u})$ is a string consisting of two kinds of letters, $\pi(\bar{c}\sigma(\boldsymbol{u})) \neq \sigma(\boldsymbol{u})$. Hence,

$$b\pi(\bar{c}\sigma(\boldsymbol{u})) \neq c\sigma(\boldsymbol{u}) \quad (\text{A}\cdot 21)$$

$$b\pi(\bar{c}\sigma(\boldsymbol{u})) \neq \bar{c}\sigma(\boldsymbol{u}) \quad (\text{A}\cdot 22)$$

hold.

From (A· 6), $c\sigma(\boldsymbol{u})$ and $\bar{c}\sigma(\boldsymbol{u})$ are form a pair of exceptional substrings by reading the substrings from right to left.

Therefore, from Assumption 2, (A· 19), (A· 20), (A· 21), and (A· 22), we have $\bar{a}\pi(c\sigma(\boldsymbol{u})) \notin \mathcal{S}_k(\tilde{\boldsymbol{x}})$ and $\bar{b}\pi(c\sigma(\boldsymbol{u})) \notin \mathcal{S}_k(\tilde{\boldsymbol{x}})$ because there exists at most one pair of exceptional substrings. Since $c\sigma(\boldsymbol{u})$ and $\bar{c}\sigma(\boldsymbol{u})$ form a pair of exceptional substrings, from condition (b) of Assumption 2, either $C(c\sigma(\boldsymbol{u})) = 1$ or $C(\bar{c}\sigma(\boldsymbol{u})) = 1$ holds. Therefore, either $C(a\pi(c\sigma(\boldsymbol{u}))) = 1$ or $C(b\pi(\bar{c}\sigma(\boldsymbol{u}))) = 1$ holds. Note that both $C(a\pi(c\sigma(\boldsymbol{u}))) = 1$ and $C(b\pi(\bar{c}\sigma(\boldsymbol{u}))) = 1$ are not satisfied because either $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ or $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$ holds, then $\bar{a}\pi(c\sigma(\boldsymbol{u})) \notin \mathcal{S}_k(\tilde{\boldsymbol{x}})$, and $\bar{b}\pi(c\sigma(\boldsymbol{u})) \notin \mathcal{S}_k(\tilde{\boldsymbol{x}})$.

Hence, if $C(a\pi(c\sigma(\boldsymbol{u}))) = 1$, then $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$. On the other hand, if $C(b\pi(\bar{c}\sigma(\boldsymbol{u}))) = 1$, then $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$.

When (A· 15) and (A· 7) hold, the proof is the same manner as in the proof when (A· 2) and (A· 7) hold. Since $\overline{C}(\bar{c}\pi(\sigma(\boldsymbol{u}))) = 0$ and $\overline{C}(\bar{c}\pi(\sigma(\bar{\boldsymbol{u}}))) = 1$, $\overline{C}_\pi(\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ from (4) and (26). Moreover, since $\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})) = \pi(\bar{c}\sigma(\boldsymbol{u}))$, $C_\pi(\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ holds. Furthermore, there exists $a \in \Sigma$ such that $\overline{C}(a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$.

We can use Lemma 6 by replacing $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))$ to $p\boldsymbol{v}$ and $\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))d$ to $\boldsymbol{v}q$ for some $d \in \Sigma$. From Lemma 6, $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))d$ is a substring of $\tilde{\boldsymbol{x}}$.

From Assumption 2, we can reconstruct $\tilde{\boldsymbol{x}}$ by using $\mathcal{S}_k(\tilde{\boldsymbol{x}})$. Therefore, there exists $\boldsymbol{s} \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ such that $\sigma(\boldsymbol{s}) = \pi(a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})))$. In other words, $\boldsymbol{s}$ is overlapping to $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))$. To determine d by using Lemma 6, when we repeat the overlapping manner from $a\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))$ for $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$, if $c\sigma(\boldsymbol{u})$ appears before $c\sigma(\bar{\boldsymbol{u}})$, then $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\bar{\boldsymbol{u}})$. On the other hand, if $c\sigma(\bar{\boldsymbol{u}})$ appears before $c\sigma(\boldsymbol{u})$, then $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\boldsymbol{u})$.

When (A· 15) and (A· 8) hold, we use the same manner as in the proof when (A· 15) and (A· 7) hold. By replacing $\bar{c}\sigma(\boldsymbol{u})$ to $c\sigma(\boldsymbol{u})$ in the proof that (A· 15) and (A· 7) hold, we can obtain $\bar{\boldsymbol{u}}$.Therefore, if $\bar{c}\sigma(\boldsymbol{u})$ appears before $\bar{c}\sigma(\bar{\boldsymbol{u}})$, then $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\bar{\boldsymbol{u}})$. On the other hand, if $\bar{c}\sigma(\bar{\boldsymbol{u}})$ appears before $\bar{c}\sigma(\boldsymbol{u})$, then $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\boldsymbol{u})$.

When (A· 15) and (A· 9) hold, $\overline{C}_\pi(\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ from the proof when (A· 15) and (A· 7) hold. Similarly, $\overline{C}_\pi(\pi(c\sigma(\bar{\boldsymbol{u}}))) = 1$.

We first consider that either $c\sigma(\bar{\boldsymbol{u}})$ or $\bar{c}\sigma(\bar{\boldsymbol{u}})$ is a string consisting of a single letter. The proof is the same manner as in the proof when (A· 4) and (A· 9) hold. If $c\sigma(\bar{\boldsymbol{u}})$ is a string consisting of a single letter, then $c\sigma(\boldsymbol{u}) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ holds. Therefore, $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$. On the other hand, if $\bar{c}\sigma(\bar{\boldsymbol{u}})$ is a string consisting of a single letter, then $\bar{c}\sigma(\boldsymbol{u}) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ holds. Therefore, $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$.

We next consider that $c\sigma(\bar{\boldsymbol{u}})$ and $\bar{c}\sigma(\bar{\boldsymbol{u}})$ are strings consisting of two kinds of letters. The proof is the same manner as in the proof when (A· 2) and (A· 7) hold. As for $c\pi(\sigma(\boldsymbol{u}))$ and $c\pi(\sigma(\bar{\boldsymbol{u}}))$, since $\overline{C}(c\pi(\sigma(\boldsymbol{u}))) = 0$ and $\overline{C}(c\pi(\sigma(\bar{\boldsymbol{u}}))) = 1$, we have $\overline{C}_\pi(\pi(c\sigma(\bar{\boldsymbol{u}}))) = 1$ from (4) and (26). Moreover, since $\pi(c\sigma(\bar{\boldsymbol{u}})) = \pi(c\sigma(\boldsymbol{u}))$, $C_\pi(\pi(c\sigma(\bar{\boldsymbol{u}}))) = 1$ holds. Furthermore, there exists $a \in \Sigma$ such that $\overline{C}(a\pi(c\sigma(\bar{\boldsymbol{u}}))) = 1$.

Similarly, as for $\bar{c}\pi(\sigma(\boldsymbol{u}))$ and $\bar{c}\pi(\sigma(\bar{\boldsymbol{u}}))$, since

$\overline{C}(\bar{c}\pi(\sigma(\boldsymbol{u}))) = 0$ and $\overline{C}(\bar{c}\pi(\sigma(\bar{\boldsymbol{u}}))) = 1$, we have $\overline{C}_\pi(\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ from (4) and (26). Moreover, since $\pi(\bar{c}\sigma(\bar{\boldsymbol{u}})) = \pi(\bar{c}\sigma(\boldsymbol{u}))$, $C_\pi(\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$ holds. Furthermore, there exists $b \in \Sigma$ such that $\overline{C}(b\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$. From (A· 15) and (A· 9), note that either $\overline{C}(a\pi(c\sigma(\bar{\boldsymbol{u}}))) = 1$ or $\overline{C}(b\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))) = 1$.

We first consider the case that $a\pi(c\sigma(\bar{\boldsymbol{u}})) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ is satisfied. The proof is the same as in the proof when (A· 2) and (A· 9) hold.

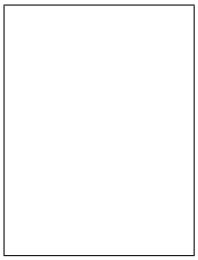
We can use Lemma 6 by replacing $a\pi(c\sigma(\bar{\boldsymbol{u}}))$ to $p\boldsymbol{v}$ and $\pi(c\sigma(\bar{\boldsymbol{u}}))d$ to $\boldsymbol{v}q$ for some $d \in \Sigma$. From Lemma 6, $a\pi(c\sigma(\bar{\boldsymbol{u}}))d$ is a substring of $\tilde{\boldsymbol{x}}$. Moreover, from (A· 9) and the fact that there exists only one substring $\bar{\boldsymbol{u}}$ having a flipping error, $C(\boldsymbol{u}) = 1$ and $C_\pi(\sigma(\boldsymbol{u})) = 1$. Since $C_\pi(\sigma(\boldsymbol{u})) = 1$, there exists $e \in \Sigma$ such that $C(\sigma(\boldsymbol{u})e) = 1$ and $C(\sigma(\boldsymbol{u})\bar{e}) = 0$. Therefore, from Lemma 6, $\boldsymbol{u}e$ is a substring of $\tilde{\boldsymbol{x}}$. Hence, $a\pi(c\sigma(\bar{\boldsymbol{u}}))de$ is a substring of $\tilde{\boldsymbol{x}}$.

From Assumption 2, we can reconstruct $\tilde{\boldsymbol{x}}$ by using $\mathcal{S}_k(\tilde{\boldsymbol{x}})$. Therefore, there exists $\boldsymbol{s} \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ such that $\sigma(\boldsymbol{s}) = \pi(a\pi(c\sigma(\bar{\boldsymbol{u}})))$. In other words, $\boldsymbol{s}$ is overlapping to $a\pi(c\sigma(\bar{\boldsymbol{u}}))$. To determine d by using Lemma 6, when we repeat the overlapping manner from $a\pi(c\sigma(\bar{\boldsymbol{u}}))$ for $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$, if $\sigma(\boldsymbol{u})f$ appears before $\sigma(\bar{\boldsymbol{u}})g$, then $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\bar{\boldsymbol{u}})$, where $f, g \in \Sigma$. On the other hand, if $\sigma(\bar{\boldsymbol{u}})g$ appears before $\sigma(\boldsymbol{u})f$, then $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\boldsymbol{u})$.

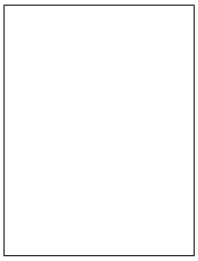
We next consider the case that $b\pi(\bar{c}\sigma(\boldsymbol{u})) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ is satisfied. We use $\bar{c}$ instead of c in the proof when $a\pi(c\sigma(\bar{\boldsymbol{u}})) \in \mathcal{S}_k(\tilde{\boldsymbol{x}})$ is satisfied. To determine d by using Lemma 6, when we repeat the overlapping manner from $b\pi(\bar{c}\sigma(\bar{\boldsymbol{u}}))$ for $\overline{\mathcal{S}}_k(\tilde{\boldsymbol{x}})$, if $\sigma(\boldsymbol{u})f$ appears before $\sigma(\bar{\boldsymbol{u}})g$ for $f, g \in \Sigma$, then $\bar{\boldsymbol{u}} = c\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\bar{\boldsymbol{u}})$. On the other hand, if $\sigma(\bar{\boldsymbol{u}})g$ appears before $\sigma(\boldsymbol{u})f$, then $\bar{\boldsymbol{u}} = \bar{c}\sigma(\bar{\boldsymbol{u}})$ and d is the last symbol of $\sigma(\boldsymbol{u})$.

Since all cases of (b) are proved, we can obtain $\bar{\boldsymbol{u}}$. Therefore, $\boldsymbol{u}$ can be obtained by flipping the last symbol of $\bar{\boldsymbol{u}}$. Hence, it completes the proof of case (b).

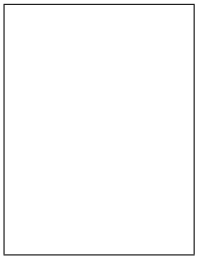
We prove all cases of (a) and (b). It completes the proof. □



Takahiro OTA received the B.E. and Ph.D. degrees from the University of Electro-Communications, Tokyo, Japan, in 1993 and 2007, respectively. In 1997, he joined Nagano Prefectural Institute of Technology, Nagano, Japan, first as a Lecturer at Department of Electronic Engineering, where from 2009, he was an Associate Professor. Since 2012, he was an Associate Professor with the Department of Computer and Systems Engineering, where from 2019, he was a Professor. In 2020, he joined at School of Network and Information, Senshu University, Tokyo, Japan, as an Associate Professor. Since 2022, he is a Professor. His current research interests are in information theory, source coding, and bio-informatics.



Keita KAMIYA received the Bachelor of Network and Inform. degree from Senshu University, Tokyo, Japan, in 2025. His research interests are in information theory and its applications.



Akiko MANADA received the M.S. degree from Tsuda College, Tokyo, Japan, in 2004, and the Ph.D. degree in Mathematics from Queen's University, Kingston, Canada, in 2009. She first worked at Claude Shannon Institute at University College Dublin, Ireland, as a postdoctoral fellow from 2009 through 2011. She then worked at the University of Electro-Communications, Japan, as an assistant professor from 2012 through 2018, and at Shonan Institute of Technology, Japan, as a lecturer from 2018 through 2021. Since 2021, she has been an associate professor at Nagaoka University of Technology, Japan. Her research interests are discrete mathematics (especially in graph theory) and its applications towards coding theory.