Error-Correcting Codes for Nanopore Sequencing

Anisha Banerjee*, Yonatan Yehezkeally*, Antonia Wachter-Zeh*, and Eitan Yaakobi†
*Institute for Communications Engineering, Technical University of Munich (TUM), Munich, Germany
†Department of Computer Science, Technion - Israel Institute of Technology, Haifa 3200003, Israel
Email: anisha.banerjee@tum.de, yonatan.yehezkeally@tum.de, antonia.wachter-zeh@tum.de, yaakobi@cs.technion.ac.il

***Abstract*—Nanopore sequencers, being superior to other sequencing technologies for DNA storage in multiple aspects, have attracted considerable attention in recent times. Their high error rates however demand thorough research on practical and efficient coding schemes to enable accurate recovery of stored data. To this end, we consider a simplified model of a nanopore sequencer inspired by Mao *et al.*, that incorporates intersymbol interference and measurement noise. Essentially, our channel model passes a sliding window of length ℓ over an input sequence, that outputs the L_1-weight of the enclosed ℓ bits and shifts by δ positions with each time step. The resulting $(\ell+1)$-ary vector, termed the *read vector*, may also be corrupted by t substitution errors. By employing graph-theoretic techniques, we deduce that for $\delta = 1$, at least $\log\log n$ bits of redundancy are required to correct a single ($t = 1$) substitution. Finally for $\ell \geq 3$, we exploit some inherent characteristics of read vectors to arrive at an error-correcting code that is optimal up to an additive constant for this setting.**

I. Introduction

The advent of DNA storage as an encouraging solution to our ever-increasing storage requirements has spurred significant research to develop superior synthesis and sequencing technologies. Among the latter, nanopore sequencing [1–3] appears to be a strong contender due to low cost, better portability and support for longer reads. In particular, this sequencing process comprises transmigrating a DNA fragment through a microscopic pore that holds ℓ nucleotides at each time instant, and measuring the variations in the ionic current, which are influenced by the different nucleotides passing through. However, due to the physical aspects of this process, multiple kinds of distortions corrupt the readout. Firstly, the simultaneous presence of $\ell > 1$ nucleotides in the pore makes the observed current dependent on multiple nucleotides instead of just one, thus causing inter-symbol interference (ISI). Next, the passage of the DNA fragment through the pore is often irregular and may involve backtracking or skipping a few nucleotides, thereby leading to duplications or deletions respectively. Furthermore, the measured current is accompanied by random noise, which might result in substitution errors.

This work was supported in part by the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation programme under Grant 801434, and in part by the European Union (ERC, DNAStorage, 865630), and in part by NSF Grant CCF 2212437. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. The work of Yonatan Yehezkeally was supported through a Carl Friedrich von Siemens Post-Doctoral Research Fellowship of the Alexander von Humboldt Foundation.

Several attempts towards developing a faithful mathematical model for the nanopore sequencer have already been made. In particular, [4] proposed a channel model that embodies the effects of ISI, deletions and random noise, while also establishing upper bounds on the capacity of this channel. The authors of [5] focused on a more deterministic model that incorporates ISI and developed an algorithm to compute its capacity. Efficient coding schemes for this abstracted channel were also suggested. More recently, a finite-state Markov channel (FSMC)-based approach was adopted to formulate a model that accounts for ISI, duplications and noisy measurements [6].

In this work, we adopt a specific variation of the model proposed in [4], that is also interesting owing to its resemblance with the transverse-read channel [7], which is relevant to racetrack memories. Specifically, we operate in the binary alphabet, i.e., consider the input symbols to be binary instead of quaternary as in DNA, and represent the process of nanopore sequencing as the concatenation of three channels as depicted in Fig. 1. The ISI channel, parameterized by (ℓ, δ), is meant to reflect the dependence of the current variations on the ℓ consecutive nucleotides that are in the pore at any given time. We may view this stage as a sliding window of size ℓ passing through an input sequence and shifting by δ positions after each time instant, thereby producing a sequence of strings of ℓ consecutive symbols, or *ℓ-mers*. Next, the substitution channel captures the effect of random noise by introducing possible substitution errors into the sequence of ℓ-mers. Finally, this erroneous sequence of ℓ-mers is converted by a memoryless channel into a sequence of discrete voltage levels according to a deterministic function, specifically the L_1-weight.

The objective of this work is to design efficient error-correcting codes for nanopore sequencing. More specifically, the aforementioned channel model is treated in the case wherein at most one substitution occurs and $\delta = 1$. The problem is stated more formally as follows.

Let $\mathcal{R}_{\ell,\delta}(\boldsymbol{x})$ represent the channel output for an input $\boldsymbol{x} \in \Sigma_2^n$, given that no substitution affected the ℓ-mers. According to our model, this is expressible as

$$\mathcal{R}_{\ell,\delta}(\boldsymbol{x}) = (\mathrm{wt}(\boldsymbol{x}_{\delta-\ell+1}^{\delta}), \mathrm{wt}(\boldsymbol{x}_{2\delta-\ell+1}^{2\delta}), \ldots, \mathrm{wt}(\boldsymbol{x}_{n-\delta+1}^{n+\ell-\delta})),$$

where for any $i \notin [n]$, we let $x_i = 0$. Now we seek to find a code $\mathcal{C} \subset \Sigma_2^n$ such that for any $\boldsymbol{c}_1, \boldsymbol{c}_2 \in \mathcal{C}$, the Hamming distance between $\mathcal{R}_{\ell,\delta}(\boldsymbol{c}_1)$ and $\mathcal{R}_{\ell,\delta}(\boldsymbol{c}_2)$ strictly exceeds 2. In other words, one can uniquely deduce the channel input despite ISI and the subsequent occurrence of at most one substitution, provided it belongs to the code $\mathcal{C}$. As a first step,

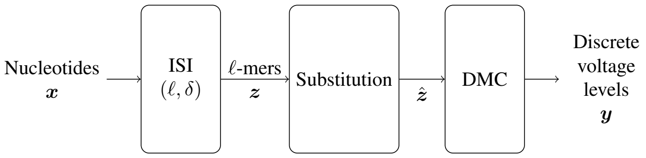

Fig. 1. Simplified model of a nanopore sequencer

we determine the minimum redundancy required by $\mathcal{C}$ with the assistance of methods outlined in [8, 9]. Following this, an optimal instantiation of $\mathcal{C}$ is proposed.

II. Preliminaries

A. Notations & Terminology

In the following, we let Σ_q indicate the q-ary alphabet $\{0, 1, \ldots, q-1\}$. Additionally, $[n]$ is used to denote the set $\{1, 2, \ldots, n\}$. Element-wise modulo operation on a vector, say $\boldsymbol{y} \in \Sigma_q^n$, is represented as

$$\boldsymbol{y} \bmod a = \big(y_1 \bmod a, y_2 \bmod a, \ldots, y_n \bmod a\big).$$

For any vector $\boldsymbol{x} = (x_1, \ldots, x_n)$, we refer to its substring $(x_i, x_{i+1}, \ldots, x_j)$ as $\boldsymbol{x}_i^j$. The L_1-weight of a vector $\boldsymbol{x}$ is denoted by $\mathrm{wt}(\boldsymbol{x}) = \sum_{i=1}^n x_i$. We also extensively use the Hamming distance, which is defined for any two vectors $\boldsymbol{x}, \boldsymbol{y} \in \Sigma_q^n$ as

$$d_H(\boldsymbol{x}, \boldsymbol{y}) = |\{i : i \in [n], x_i \neq y_i\}|.$$

Throughout this paper, we focus on the case of $q = 2$ and assume existence of integers n, ℓ, and δ that satisfy the relation $(n+\ell) \bmod \delta = 0$. In this framework, the (ℓ, δ)-*read vector* of any $\boldsymbol{x} \in \Sigma_2^n$ is of length $(n+\ell)/\delta - 1$ and is denoted by

$$\mathcal{R}_{\ell,\delta}(\boldsymbol{x}) = (\mathrm{wt}(\boldsymbol{x}_{\delta-\ell+1}^{\delta}), \mathrm{wt}(\boldsymbol{x}_{2\delta-\ell+1}^{2\delta}), \ldots, \mathrm{wt}(\boldsymbol{x}_{n-\delta+1}^{n+\ell-\delta})),$$

where for any $i \notin [n]$, we let $x_i = 0$.

Remark: The above definition of an (ℓ, δ)-read vector appears similar to that of the (ℓ, δ)-transverse-read vector introduced in [7], except that $\mathcal{R}_{\ell,\delta}(\boldsymbol{x})$ begins and ends with the L_1-weights of substrings $\boldsymbol{x}_1^{\delta}$ and $\boldsymbol{x}_{n-\delta+1}^{n}$ respectively, even though its intermediate elements signify L_1-weights of length-ℓ substrings. This is motivated by the fact that we obtain a current reading even when the DNA strand has only entered the nanopore partially.

Next, we denote the i-th element of $\mathcal{R}_{\ell,\delta}(\boldsymbol{x})$ by $\mathcal{R}_{\ell,\delta}(\boldsymbol{x})_i$, i.e., $\mathcal{R}_{\ell,\delta}(\boldsymbol{x})_i = \mathrm{wt}(\boldsymbol{x}_{i\delta-\ell+1}^{i\delta})$. Another useful notation, designated as the i-th *read sub-derivative*, represents a specific subsequence of the derivative of $\mathcal{R}_{\ell,\delta}(\boldsymbol{x})$, and is defined for any $\alpha \in \Sigma_{\lfloor \frac{\ell}{\delta} \rfloor}$ as

$$\begin{aligned}\Delta_{\ell,\delta}^{\alpha}(\boldsymbol{x}) &= (\mathcal{R}(\boldsymbol{x})_{\alpha+1} - \mathcal{R}(\boldsymbol{x})_{\alpha}, \mathcal{R}(\boldsymbol{x})_{\alpha+\lfloor \frac{\ell}{\delta} \rfloor+1} - \mathcal{R}(\boldsymbol{x})_{\alpha+\lfloor \frac{\ell}{\delta} \rfloor}, \\ &\qquad \ldots, \mathcal{R}(\boldsymbol{x})_{\alpha+k\lfloor \frac{\ell}{\delta} \rfloor+1} - \mathcal{R}(\boldsymbol{x})_{\alpha+k\lfloor \frac{\ell}{\delta} \rfloor}) \\ &= (\mathrm{wt}(\boldsymbol{x}_{\alpha\delta+1}^{(\alpha+1)\delta}) - \mathrm{wt}(\boldsymbol{x}_{\alpha\delta-\ell+1}^{(\alpha+1)\delta-\ell}), \ldots \\ &\qquad \ldots, \mathrm{wt}(\boldsymbol{x}_{(\alpha+k\lfloor \frac{\ell}{\delta} \rfloor)\delta+1}^{(\alpha+k\lfloor \frac{\ell}{\delta} \rfloor+1)\delta}) - \mathrm{wt}(\boldsymbol{x}_{(\alpha+k\lfloor \frac{\ell}{\delta} \rfloor)\delta-\ell+1}^{(\alpha+k\lfloor \frac{\ell}{\delta} \rfloor+1)\delta-\ell})),\end{aligned}$$

where $k = \lfloor \frac{n+\ell-(\alpha+1)\delta}{\delta\lfloor \ell/\delta \rfloor} \rfloor$ and for any $p \notin [\frac{n+\ell-\delta}{\delta}]$ and $m \notin [n]$, we let $\mathcal{R}(\boldsymbol{x})_p = 0$ and $x_m = 0$. When clear from the context, ℓ and δ will be removed from the preceding notations.

Example 1. *Consider* $\boldsymbol{x} = (1, 0, 1, 1, 0, 0)$. *The* $(3, 1)$*-read vector of* $\boldsymbol{x}$ *is thus* $\mathcal{R}_{3,1}(\boldsymbol{x}) = (1, 1, 2, 2, 2, 1, 0, 0)$. *Evidently,* $\mathcal{R}_{3,1}(\boldsymbol{x})_3 = 2$, $\Delta_{3,1}^0(\boldsymbol{x}) = (1, 0, -1)$, $\Delta_{3,1}^1(\boldsymbol{x}) = (0, 0, 0)$ *and* $\Delta_{3,1}^2(\boldsymbol{x}) = (1, -1, 0)$.

A straightforward extension of the preceding definitions for $q > 2$ involves replacing the L_1-weights of substrings with their respective compositions. Here, the composition of any $\boldsymbol{x} \in \Sigma_q^n$ signifies $c(\boldsymbol{x}) = 0^{i_0} 1^{i_1} \cdots (q-2)^{i_{q-2}} (q-1)^{n-\sum_{j=0}^{q-2} i_j}$, if $\boldsymbol{x}$ constitutes i_0 0s, i_1 1s and so on.

As mentioned earlier, [7] investigated a similar model designated as the transverse-read channel, in connection with racetrack memories. Therein, the information limit of this channel was derived for different parameters and several codes enabling unique reconstruction were proposed. For $\ell = 2$ and $\delta = 1$, certain error-correcting codes were also presented.

B. Properties of the Read Vectors

By the nature of its definition, each valid read vector possesses certain properties which often enable us to detect errors and thereby assist in designing error-correcting constructions of improved redundancies. These are summarized below.

Proposition 1. *P1 For any* $i \in [\frac{n+\ell-2\delta}{\delta}]$, *it holds that* $|\mathcal{R}(\boldsymbol{x})_{i+1} - \mathcal{R}(\boldsymbol{x})_i| \leq \delta$.

- *P2 Let two vectors* $\boldsymbol{x}, \boldsymbol{y} \in \Sigma_2^n$ *be such that* $d_H(\boldsymbol{x}, \boldsymbol{y}) = 1$. *If* $\ell \bmod \delta = 0$, *then the* (ℓ, δ)*-read vectors* $\mathcal{R}(\boldsymbol{x})$ *and* $\mathcal{R}(\boldsymbol{y})$ *satisfy* $d_H(\mathcal{R}(\boldsymbol{x}), \mathcal{R}(\boldsymbol{y})) = \frac{\ell}{\delta}$.
- *P3 For any* ℓ, δ *and* $\boldsymbol{x} \in \Sigma_2^n$ *such that* $\ell \bmod \delta = 0$, *it holds that* $\sum_{i=1}^{\frac{n+\ell-\delta}{\delta}} \mathcal{R}(\boldsymbol{x})_i = \frac{\ell}{\delta}\mathrm{wt}(\boldsymbol{x})$.
- *P4 If* $\delta = 1$, *then for any* $\boldsymbol{x} \in \Sigma_2^n$, $\boldsymbol{x}$ *can be directly inferred from the first or last* n *elements of either* $\mathcal{R}(\boldsymbol{x})$ *or* $\mathcal{R}(\boldsymbol{x}) \bmod 2$.
- *P5 If* $\ell \bmod \delta = 0$, *then for all* $\alpha \in \Sigma_{\frac{\ell}{\delta}}$ *and* $\boldsymbol{x} \in \Sigma_2^n$, *it holds that* $\mathrm{wt}(\Delta_{\ell,\delta}^{\alpha}(\boldsymbol{x})) = 0$.
- *P6 For any* $\alpha \in \Sigma_{\ell}$ *and* $\delta = 1$, *if we denote the non-zero elements of* $\Delta^{\alpha}(\boldsymbol{x})$, *in order, by* $d_0, d_1, \ldots, d_D$, *where* $D \leq \frac{n+2\ell-\alpha-1}{\ell}$, *then* $d_0 = 1$ *and for any* $i \in \Sigma_D$, $\{d_i, d_{i+1}\} = \{-1, 1\}$.
- *P7 If* $\ell \bmod \delta = 0$, *then for any* $\boldsymbol{x} \in \Sigma_2^n$ *and* $\alpha \in \Sigma_{\frac{\ell}{\delta}}$, *the cumulative sum of the first* $m+1$ *elements of* $\Delta_{\ell,\delta}^{\alpha}(\boldsymbol{x})$ *is* $\mathrm{wt}(\boldsymbol{x}_{m\ell+\alpha\delta+1}^{m\ell+(\alpha+1)\delta})$[1], *i.e.,* $\sum_{j=0}^{m} \big(\mathcal{R}(\boldsymbol{x})_{\alpha+j\ell+1} - \mathcal{R}(\boldsymbol{x})_{\alpha+j\ell}\big) = \mathrm{wt}(\boldsymbol{x}_{m\ell+\alpha\delta+1}^{m\ell+(\alpha+1)\delta})$. *Thus,* $\Delta_{\ell,\delta}^{\alpha}(\boldsymbol{x})$ *determines* $(\mathrm{wt}(\boldsymbol{x}_{\alpha\delta+1}^{(\alpha+1)\delta}), \mathrm{wt}(\boldsymbol{x}_{\ell+\alpha\delta+1}^{\ell+(\alpha+1)\delta}), \ldots)$, *which in the special case of* $\delta = 1$, *is effectively* $(x_{\alpha+1}, x_{\alpha+\ell+1}, \ldots)$.

P4 can be verified through attempts to reconstruct $\boldsymbol{x}$ from $\mathcal{R}(\boldsymbol{x})$ or $\mathcal{R}(\boldsymbol{x}) \bmod 2$, sequentially from left to right (or right to left). One may similarly verify P7 by attempting reconstruction using $\Delta_{\ell,1}^{\alpha}(\boldsymbol{x})$, while the remaining properties result directly from the definitions in Section II-A.

Example 2. *Given* $\mathcal{R}_{3,1}(\boldsymbol{x}) \bmod 2 = (1, 1, 0, 0, 0, 1, 0, 0)$ *from Example 1, we wish to reconstruct* $\boldsymbol{x}$. *Firstly, we observe that* $\mathcal{R}(\boldsymbol{x})_1 = x_1 = 1$. *Next,* $\mathcal{R}(\boldsymbol{x})_2 \bmod 2 = x_1 \oplus x_2 = 1$, *causing* $x_2 = 0$. *Such a left-to-right reconstruction of*

[1] Analogous result exists for sum of last $m+1$ elements.

$\mathcal{R}(\boldsymbol{x}) \bmod 2$ *leads us to* $\boldsymbol{x} = (1,0,1,1,0,0)$*, as in Example 1. Right-to-left reconstruction will yield the same result.*

A natural consequence of such left-to-right and right-to-left reconstruction processes is the following.

Corollary 1. *If* $\delta = 1$*, then for any* $\boldsymbol{x} \in \Sigma_2^n$*,* $\boldsymbol{x}_i^j$ *can be uniquely determined, either from*

1) $\boldsymbol{x}_{i-\ell+1}^{i-1}$ *and* $(\mathcal{R}(\boldsymbol{x})_i, \mathcal{R}(\boldsymbol{x})_{i+1}, \ldots, \mathcal{R}(\boldsymbol{x})_j)$*; or*
2) $\boldsymbol{x}_{j+1}^{j+\ell-1}$ *and* $(\mathcal{R}(\boldsymbol{x})_{i+\ell-1}, \mathcal{R}(\boldsymbol{x})_{i+\ell}, \ldots, \mathcal{R}(\boldsymbol{x})_{j+\ell-1})$*,*

where for all $k \notin [n]$*,* $x_k = 0$*.*

Another important consequence of the aforementioned properties is stated below.

Lemma 1. *When* $\ell > 1$ *and* $\delta = 1$*, for any two distinct* $\boldsymbol{x}, \boldsymbol{y} \in \Sigma_2^n$*,* $d_H(\mathcal{R}(\boldsymbol{x}), \mathcal{R}(\boldsymbol{y})) \geq 2$*.*

Proof: Assume that $d_H(\mathcal{R}(\boldsymbol{x}), \mathcal{R}(\boldsymbol{y})) = 1$, and let i denote the index where $\mathcal{R}(\boldsymbol{x})$ and $\mathcal{R}(\boldsymbol{y})$ differ, i.e., $\mathcal{R}(\boldsymbol{x})_i \neq \mathcal{R}(\boldsymbol{y})_i$. From P3, we infer that

$$\mathrm{wt}(\mathcal{R}(\boldsymbol{x})) - \mathrm{wt}(\mathcal{R}(\boldsymbol{y})) = (\mathcal{R}(\boldsymbol{x})_i - \mathcal{R}(\boldsymbol{y})_i) = 0 \pmod{\ell}.$$

Since $\mathcal{R}(\boldsymbol{x})_i, \mathcal{R}(\boldsymbol{y})_i \in \Sigma_{\ell+1}$, the only possibility involves $\{\mathcal{R}(\boldsymbol{x})_i, \mathcal{R}(\boldsymbol{y})_i\} = \{0, \ell\}$. Due to $\ell > 1$, we have that $\Delta^{i \bmod \ell}(\boldsymbol{x}), \Delta^{i \bmod \ell}(\boldsymbol{y})$ differ in a unique index, and their difference at that index equals $\pm\ell$, which contradicts P5 for at least one of $\boldsymbol{x}, \boldsymbol{y}$. ■

C. Error Model

Similar to [7], we study the occurrence of substitution errors in read vectors and design suitable error-correcting constructions. Specifically, a code is said to be a *t-substitution (ℓ, δ)-read code* if for any $\boldsymbol{x}, \boldsymbol{y}$ that belong to this code, it holds that $d_H(\mathcal{R}(\boldsymbol{x}), \mathcal{R}(\boldsymbol{y})) > 2t$.

In this work, we focus on the case when $\delta = 1$ and $t = 1$. To this end, we seek to find a code that can correct a single substitution error in the read vectors of its constituent codewords, i.e., a single-substitution $(\ell, 1)$-read code. In the upcoming sections, we endeavor to derive an upper bound on the cardinality of such a code, and subsequently propose an optimal instantiation of the same.

III. Minimum Redundancy of Single-substitution $(\ell, 1)$-read codes

To establish a lower bound on the redundancy required by a single-substitution $(\ell, 1)$-read code, we first attempt to characterize the relationship between any two binary vectors $\boldsymbol{x}, \boldsymbol{y} \in \Sigma_2^n$, that might be confusable after a single substitution in their respective read vectors.

A. Characterization of confusable read vectors

To proceed in this direction, we first note from Lemma 1 that there exists no two distinct vectors $\boldsymbol{x}, \boldsymbol{y} \in \Sigma_2^n$ that satisfy $d_H(\mathcal{R}(\boldsymbol{x}), \mathcal{R}(\boldsymbol{y})) = 1$ for any $\ell > 1$. Thus, we attempt to ascertain the conditions under which $d_H(\mathcal{R}(\boldsymbol{x}), \mathcal{R}(\boldsymbol{y})) = 2$ may occur.

Theorem 1. *For* $\ell \geq 3$ *and any* $\boldsymbol{x}, \boldsymbol{y} \in \Sigma_2^n$*, the following are equivalent:*

1) $d_H(\mathcal{R}(\boldsymbol{x}), \mathcal{R}(\boldsymbol{y})) = 2$*.*
2) *There exist distinct* $i, j \in [n + \ell - 1]$*,* $j = i \pmod{\ell}$*, such that* $\mathcal{R}(\boldsymbol{x})_i - \mathcal{R}(\boldsymbol{y})_i = \mathcal{R}(\boldsymbol{y})_j - \mathcal{R}(\boldsymbol{x})_j = 1$ *and* $\mathcal{R}(\boldsymbol{x})_r = \mathcal{R}(\boldsymbol{y})_r$ *for all* $r \notin \{i, j\}$*.*
3) *There exist* $p \geq 1$ *and* $i \in [n - (p-1)\ell - 1]$ *such that for all* $m \in \Sigma_p$ *it holds that* $\boldsymbol{x}_{i+m\ell}^{i+m\ell+1} = (1,0)$*,* $\boldsymbol{y}_{i+m\ell}^{i+m\ell+1} = (0,1)$ *(or vice versa), and* $x_r = y_r$ *for all* $r \notin \bigcup_{m \in \Sigma_p} \{i + m\ell, i + m\ell + 1\}$*.*

Further, if these conditions hold, then $j = i + p\ell$ *in the above notation.*

B. Upper bound on code size

We derive a lower bound on the redundancy required by a single-substitution $(\ell, 1)$-read code, by adopting the approach employed in [8, 9]. More precisely, we consider a graph $\mathcal{G}(n)$ containing vertices corresponding to all vectors in Σ_2^n. Any two vertices in $\mathcal{G}(n)$ that signify two distinct binary vectors, say $\boldsymbol{x}, \boldsymbol{y} \in \Sigma_2^n$, are considered to be adjacent if and only if $d_H(\mathcal{R}(\boldsymbol{x}), \mathcal{R}(\boldsymbol{y})) = 2$. Therefore, any subset of vertices of $\mathcal{G}(n)$, wherein no two vertices are adjacent, is a 1-substitution $(\ell, 1)$-read code.

Definition 1. *A* ***clique cover*** $\mathcal{Q}$ *is a collection of cliques in a graph* $\mathcal{G}$*, such that every vertex in* $\mathcal{G}$ *belongs to at least one clique in* $\mathcal{Q}$*.*

The following graph-theoretic result is well-known [8, 10].

Theorem 2. *If* $\mathcal{Q}$ *is a clique cover, then the size of any independent set is at most* $|\mathcal{Q}|$*.*

For the remainder of this section, we seek to define a clique cover $\mathcal{Q}$ by utilizing Theorem 1. By virtue of Theorem 2, the size of such a clique cover will serve as an upper bound on the cardinality of a 1-substitution $(\ell, 1)$-read code.

Definition 2. *[8, Sec. III] Let* $\mathcal{G}'(n)$ *be the graph whose vertices are all vectors in* Σ_2^n*, and an edge connects* $\boldsymbol{x}, \boldsymbol{y} \in \Sigma_2^n$ *if and only if* $\{\boldsymbol{x}, \boldsymbol{y}\} = \{\boldsymbol{u} \circ (01)^j \circ \boldsymbol{v}, \boldsymbol{u} \circ (10)^j \circ \boldsymbol{v}\}$*, for some* j *and sub-strings* $\boldsymbol{u}, \boldsymbol{v}$*.*

Our method of proof would be to pull-back a clique cover from $\mathcal{G}'$, based on [8, Lem. 10], into $\mathcal{G}$. In order to do that, we have the following definition:

Definition 3. *For a positive integer p, define a permutation* π_p *on* Σ_2^n *as follows. For all* $\boldsymbol{x} \in \Sigma_2^n$*, arrange the coordinates of* $\boldsymbol{x}_1^{p\ell\lfloor n/(p\ell)\rfloor}$ *in a matrix* $X \in \Sigma^{p\lfloor n/(p\ell)\rfloor \times \ell}$*, by row (first fill the first row from left to right, then the next, etc.). Next, partition* X *into sub-matrices of dimension* $p \times 2$ *(if* ℓ *is odd, we ignore* X*'s right-most column). Finally, going through each sub-matrix (from left to right, and then top to bottom), we concatenate its* rows*, to obtain* $\pi_p(\boldsymbol{x})$ *(where unused coordinates from* $\boldsymbol{x}$ *are appended arbitrarily).*

More precisely, for all $0 \leq i < \lfloor \frac{n}{p\ell} \rfloor$*,* $0 \leq j < \lfloor \frac{\ell}{2} \rfloor$ *and* $0 \leq k < p$ *denote*

$$\boldsymbol{x}^{(i,j,k)} = x_{(ip+k)\ell+2j+1} x_{(ip+k)\ell+2j+2};$$

then

$$\boldsymbol{x}^{(i,j)} = \boldsymbol{x}^{(i,j,0)} \circ \cdots \circ \boldsymbol{x}^{(i,j,p-1)}$$

and

$$\boldsymbol{x}^{(i)} = \boldsymbol{x}^{(i,0)} \circ \cdots \circ \boldsymbol{x}^{(i,\lfloor \ell/2 \rfloor -1)}.$$

Then $\pi_p(\boldsymbol{x}) = \boldsymbol{x}^{(0)} \circ \cdots \circ \boldsymbol{x}^{(\lfloor n/p\ell \rfloor -1)} \circ \tilde{\boldsymbol{x}}$, *where* $\tilde{\boldsymbol{x}}$ *is composed of all coordinates of* $\boldsymbol{x}$ *not earlier included.*

Example 3. $\boldsymbol{x} = (1,0,1,1,0,0)$ *and* $\boldsymbol{y} = (0,1,1,0,1,0)$ *satisfy* $d_H(\mathcal{R}_{3,1}(\boldsymbol{x}), \mathcal{R}_{3,1}(\boldsymbol{y})) = 2$. *To obtain* $\pi_p(\boldsymbol{x})$ *and* $\pi_p(\boldsymbol{y})$ *for* $p = 2$, *note that*

$$X = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \; Y = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

Since ℓ *is odd, we ignore the last column in* X *and* Y, *and partition the respective results into* 2×2 *sub-matrices to ultimately obtain* $\pi_p(\boldsymbol{x}) = (1,0,1,0,1,0)$ *and* $\pi_p(\boldsymbol{y}) = (0,1,0,1,1,0)$ *(here, unused coordinates were appended in the order of their indices).*

Definition 4. *[8, Sec. III] For a positive integer* p, *let*

$$\Lambda_p = \left\{ (\boldsymbol{a})^j (\boldsymbol{b})^{p-j} : j \in [p], \{\boldsymbol{a}, \boldsymbol{b}\} = \{01, 10\} \right\}$$

where $\boldsymbol{a}^0 = \boldsymbol{b}^0$ *is the empty word, and* $\tilde{\Lambda}_p = \Sigma_2^{2p} \setminus \Lambda_p$. *Further, let*

$$\Gamma = \left\{ (\boldsymbol{u}, \boldsymbol{w}) : i \in [m], \boldsymbol{u} \in \tilde{\Lambda}_p^{i-1}, \boldsymbol{w} \in \Sigma_2^{2p(m-i)} \right\},$$

where $m = \lfloor \frac{\ell}{2} \rfloor \lfloor \frac{n}{p\ell} \rfloor$, *and* $\tilde{\Lambda}_p^0$ *is the singleton containing an empty word. Then, for all* $(\boldsymbol{u}, \boldsymbol{w}) \in \Gamma$ *define*

$$Q^{(0)}_{(\boldsymbol{u},\boldsymbol{w})} = \left\{ \boldsymbol{u}(01)^h (10)^{p-h} \boldsymbol{w} : h \in [p] \right\},$$
$$Q^{(1)}_{(\boldsymbol{u},\boldsymbol{w})} = \left\{ \boldsymbol{u}(10)^h (01)^{p-h} \boldsymbol{w} : h \in [p] \right\}.$$

Finally, let

$$\mathcal{Q}(m,p) = \left\{ \{\boldsymbol{x}\} : \boldsymbol{x} \in \tilde{\Lambda}_p^m \right\} \cup \left\{ Q^{(0)}_\gamma, Q^{(1)}_\gamma : \gamma \in \Gamma \right\}.$$

Lemma 2. *[8, Lem. 10]* $\mathcal{Q}(m,p)$ *is a clique-cover of* $\mathcal{G}'(2pm)$, *where* $m = \lfloor \frac{\ell}{2} \rfloor \lfloor \frac{n}{p\ell} \rfloor$.

Theorem 3. *Let*

$$\mathcal{Q}_p = \left\{ \pi_p^{-1}(Q \times \{\boldsymbol{z}\}) : Q \in \mathcal{Q}(m,p), \boldsymbol{z} \in \Sigma_2^{n-2pm} \right\},$$

where $\pi_p^{-1}(A) = \{\boldsymbol{u} \in \Sigma_2^n : \pi_p(\boldsymbol{u}) \in A\}$. *Then* $\mathcal{Q}_p$ *is a clique-cover in* $\mathcal{G}(n)$.

Proof: First, observe that it readily follows from $\bigcup \mathcal{Q}(m,p) = \Sigma_2^{2pm}$ that $\bigcup \mathcal{Q}_p = \Sigma_2^n$. It is therefore left to prove that every element of $\mathcal{Q}_p$ is a clique of $\mathcal{G}(n)$.

Then, observe for all $Q \in \mathcal{Q}(m,p)$ and $\boldsymbol{z} \in \Sigma_2^{n-2pm}$ that either Q is a singleton, or all elements $\boldsymbol{y} \in Q \times \{\boldsymbol{z}\}$ agree on all coordinates y_k except $2(i-1)p < k \leq 2ip$ for some $i \in [m]$, and $\boldsymbol{y}_{2(i-1)p}^{2ip} \in \left\{ (01)^h (10)^{p-h}, (10)^h (01)^{p-h} \right\}$ for some $h \in [p]$. I.e., either $\pi_p^{-1}(Q \times \{\boldsymbol{z}\})$ is a singleton, or all elements $\boldsymbol{x} \in \pi_p^{-1}(Q \times \{\boldsymbol{z}\})$ agree on all coordinates except, in the notation of Definition 3, $\boldsymbol{x}^{(i,j)}$ for some $0 \leq i < \lfloor \frac{n}{p\ell} \rfloor$, $0 \leq j < \lfloor \frac{\ell}{2} \rfloor$, and $\boldsymbol{x}^{(i,j)} \in \left\{ (01)^h (10)^{p-h}, (10)^h (01)^{p-h} \right\}$ for some $h \in [p]$. That is, $\boldsymbol{x}^{(i,j,k)} = 01$ (10) for all $0 \leq k < h$, and $\boldsymbol{x}^{(i,j,k)} = 10$ (respectively, 01) for all $h \leq k < p$. By Theorem 1, it holds that $d_H(\mathcal{R}(\boldsymbol{x}_1), \mathcal{R}(\boldsymbol{x}_2)) = 2$ for all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \pi_p^{-1}(Q \times \{\boldsymbol{z}\})$. ■

Finally, we can obtain a lower bound on the redundancy of a single-substitution $(\ell, 1)$-read code from the following result.

Lemma 3. *[8, Lem. 12]*

$$|\mathcal{Q}(m,p)| = 2^{2pm} \left[\left(1 - \frac{2p}{2^{2p}}\right)^m + \frac{1}{p}\left(1 - \left(1 - \frac{2p}{2^{2p}}\right)^m\right) \right],$$

where $m = \lfloor \frac{\ell}{2} \rfloor \lfloor \frac{n}{p\ell} \rfloor$.

It readily follows that for any positive integer p,

$$\log_2 |\mathcal{Q}_p| = n - \log_2(p) + \log_2 \left(1 + (p-1)\left(1 - \frac{2p}{2^{2p}}\right)^m \right).$$

Based on $m \geq \lfloor \frac{n}{2p} \rfloor - \lfloor \frac{\ell}{2} \rfloor$ we may further bound

$$\log_2 |\mathcal{Q}_p| \leq n - \log_2(p) + p\left(1 - \frac{2p}{2^{2p}}\right)^{\lfloor n/2p \rfloor} \Big/ \left(1 - \frac{2p}{2^{2p}}\right)^{\lfloor \ell/2 \rfloor}.$$

It was shown in [8] that letting $p = \lceil \frac{1}{2}(1-\epsilon)\log_2(n) \rceil$ for any $0 < \epsilon < 1$ yields $p\left(1 - \frac{2p}{2^{2p}}\right)^{\lfloor n/2p \rfloor} = o(1)$, hence based on Theorem 2 we arrive at the following theorem.

Theorem 4. *The redundancy of a* 1*-substitution* $(\ell, 1)$*-read code is bounded from below by*

$$\log_2 \log_2(n) - \log_2\left(\tfrac{2}{1-\epsilon}\right) - o(1).$$

IV. Single Substitution Read Codes

It is already implied by P4 that a redundancy of $t \log n$ bits suffices to correct at most t substitutions in the $(\ell, 1)$-read vector. However according to Theorem 4, a more efficient code may exist for the $t = 1$ case. This section introduces such a construction that is optimal up to a constant.

For any $\boldsymbol{x} \in \Sigma_2^n$, we define a specific permutation of $\mathcal{R}_{\ell,\delta}(\boldsymbol{x})$ as

$$\mathcal{R}^\pi(\boldsymbol{x}) = \mathcal{R}^0(\boldsymbol{x}) \circ \mathcal{R}^1(\boldsymbol{x}) \circ \cdots \circ \mathcal{R}^{\ell-1}(\boldsymbol{x}),$$

where $\mathcal{R}^{i-1}(\boldsymbol{x}) = (\mathcal{R}(\boldsymbol{x})_i, \mathcal{R}(\boldsymbol{x})_{i+\ell}, \ldots, \mathcal{R}(\boldsymbol{x})_{i+\lfloor \frac{n+\ell-1-i}{\ell} \rfloor \ell})$ for all $i \in [\ell]$. In addition, we introduce the following notation to represent the concatenation of all i-th read sub-derivatives.

$$\Delta(\boldsymbol{x}) = \Delta^0(\boldsymbol{x}) \circ \Delta^1(\boldsymbol{x}) \circ \cdots \circ \Delta^{\lfloor \frac{\ell}{\delta} \rfloor -1}(\boldsymbol{x}).$$

To simplify presentation, we also define the following.

Definition 5. *Let* $RLL(a)$ *be the set of all finite-length binary vectors whose runs of* 0*s are of length at most* a.

Definition 6. *For* $n, a > 0$, *let* $\mathcal{H}(n,a)$ *be the binary linear code of length* n, *defined by the parity-check matrix*

$$\underbrace{\begin{bmatrix} \mathbf{H}_a & \mathbf{H}_a & \cdots & \mathbf{H}_a \end{bmatrix}}_{\frac{n}{2^a-1} \text{ times}},$$

where $\mathbf{H}_a$ *represents the parity-check matrix of a Hamming code of order* a, *i.e.,* $\mathbf{H}_a$ *contains all non-zero binary length-*a *vectors as its columns.*

Finally, we propose the following code to correct a single substitution in $(\ell, 1)$-read vectors for $\ell \geq 3$.

Construction 1.

$$\mathcal{C}(n,\ell) = \{\boldsymbol{x} \in \Sigma_2^n : \Delta(\boldsymbol{x}) \bmod 2 \in RLL(\log 2(n+\ell)),\ \mathcal{R}^\pi(\boldsymbol{x}) \bmod 2 \in \mathcal{H}(n+\ell-1, \log\log 8(n+\ell)+1)\}.$$

From [11, Lemma 2], we infer that this construction requires at most $\log\log n + \log\left(1 + \frac{3+\log(1+\ell/n)}{\log n}\right) + 2$ redundant bits.

To prove that $\mathcal{C}(n,\ell)$ is a 1-substitution $(\ell,1)$-read code, we first show that some error patterns are trivial to correct.

Lemma 4. *Let $\mathcal{R}(\boldsymbol{x})'$ be derived from the $(\ell,1)$-read vector $\mathcal{R}(\boldsymbol{x})$, where $\ell > 1$ and $\boldsymbol{x} \in \Sigma_2^n$, by a single substitution, and suppose $|\mathcal{R}(\boldsymbol{x})_i - \mathcal{R}(\boldsymbol{x})'_i| > 1$ for the unique $i \in [(n+\ell)/\delta - 1]$ such that $\mathcal{R}(\boldsymbol{x})_i \neq \mathcal{R}(\boldsymbol{x})'_i$. Then $\mathcal{R}(\boldsymbol{x})$ can be uniquely recovered from $\mathcal{R}(\boldsymbol{x})'$.*

Proof: Supposing that the error occurred at index k, we may express the noisy read vector as $\mathcal{R}(\boldsymbol{x})' = (\mathcal{R}(\boldsymbol{x})'_1, \ldots, \mathcal{R}(\boldsymbol{x})'_{n+\ell-1})$, where $\mathcal{R}(\boldsymbol{x})'_k \neq \mathcal{R}(\boldsymbol{x})_k$ and $\mathcal{R}(\boldsymbol{x})'_p = \mathcal{R}(\boldsymbol{x})_p$ for all $p \neq k$. It follows from $\mathcal{R}(\boldsymbol{x})_p - \mathrm{wt}(\boldsymbol{x}_{p-\ell+1}^{p-1}) = x_p \in \Sigma_2$ for all $p \in [n]$, and $\mathcal{R}(\boldsymbol{x})'_k - \mathrm{wt}(\boldsymbol{x}_{k-\ell+1}^{k-1}) = x_k + (\mathcal{R}(\boldsymbol{x})'_k - \mathcal{R}(\boldsymbol{x})_k) \notin \Sigma_2$, that k is the minimum index for which this process of left-to-right reconstruction yields a non-binary value, hence it may uniquely be identified from $\mathcal{R}(\boldsymbol{x})'$. Then, if $k \leq n$, Corollary 1 allows accurate reconstruction of $\boldsymbol{x}_k^n$ from $(\mathcal{R}(\boldsymbol{x})_{k+\ell-1}, \ldots, \mathcal{R}(\boldsymbol{x})_{n+\ell-1})$. ■

Due to Lemma 4, we focus for the rest of the section on proving that $\mathcal{C}(n,\ell)$ can correct a single substitution satisfying $|\mathrm{wt}(\mathcal{R}(\boldsymbol{x})) - \mathrm{wt}(\mathcal{R}(\boldsymbol{x})')| \leq 1$. Next, we demonstrate that the index of such substitutions may be narrowed down.

Lemma 5. *If a substitution error affects the $(\ell,1)$-read vector of some $\boldsymbol{x} \in \Sigma_2^n$ where $\ell \geq 3$, thus producing a noisy copy $\mathcal{R}(\boldsymbol{x})'$, then there exist $\alpha, \beta \in \Sigma_\ell$ where $\beta = (\alpha - 1) \bmod \ell$, such that $\mathrm{wt}(\Delta^\beta(\boldsymbol{x})') = -\mathrm{wt}(\Delta^\alpha(\boldsymbol{x})') \neq 0$, and for all $\gamma \notin \{\alpha, \beta\}$, $\mathrm{wt}(\Delta^\gamma(\boldsymbol{x})') = 0$. This implies that*

1) *the error value is $\mathrm{wt}(\Delta^\beta(\boldsymbol{x})') = -\mathrm{wt}(\Delta^\alpha(\boldsymbol{x})')$; and*
2) *the error occurred at an index $k \in [n+\ell-1]$, where $k = \alpha \pmod{\ell}$.*

Proof: Suppose that the concerned substitution error occurs at index $k \in [n+\ell-1]$. Thus, the noisy read vector can be expressed as $\mathcal{R}(\boldsymbol{x})' = (\mathcal{R}(\boldsymbol{x})'_1, \ldots, \mathcal{R}(\boldsymbol{x})'_{n-\ell+1})$, where $\mathcal{R}(\boldsymbol{x})'_k \neq \mathcal{R}(\boldsymbol{x})_k$ and $\mathcal{R}(\boldsymbol{x})'_p = \mathcal{R}(\boldsymbol{x})_p$ for all $p \neq k$.

Now observe that $\Delta^{(k-1) \bmod \ell}(\boldsymbol{x})'$ and $\Delta^{k \bmod \ell}(\boldsymbol{x})'$ no longer uphold P5. Instead,

$$\begin{aligned}\mathrm{wt}(\Delta^{(k-1) \bmod \ell}(\boldsymbol{x})') &= -\mathrm{wt}(\Delta^{k \bmod \ell}(\boldsymbol{x})') \\ &= \mathcal{R}(\boldsymbol{x})'_k - \mathcal{R}(\boldsymbol{x})_k,\end{aligned}$$

which is evidently the error value. The preceding equation suggests that the error occurred somewhere in $\mathcal{R}^{(k-1) \bmod \ell}(\boldsymbol{x})'$, which is a subsequence of $\mathcal{R}(\boldsymbol{x})'$. Alternatively, we say that the error position h satisfies $h - k = 0 \pmod{\ell}$. ■

Example 4. *$\mathcal{R}_{3,1}(\boldsymbol{v})' = (1,1,2,3,2,1,0,0)$ arises from a substitution in the $(3,1)$ read vector of some $\boldsymbol{v} \in \Sigma_2^6$. As $\mathrm{wt}(\Delta^0(\boldsymbol{v})') = -\mathrm{wt}(\Delta^1(\boldsymbol{v})') = 1$, Lemma 5 suggests that the error has value 1 and occurred somewhere in $(\mathcal{R}(\boldsymbol{v})'_1, \mathcal{R}(\boldsymbol{v})'_4, \mathcal{R}(\boldsymbol{v})'_7)$. Now assigning $\mathcal{R}(\boldsymbol{v})'_1 \leftarrow \mathcal{R}(\boldsymbol{v})'_1 - 1$ or $\mathcal{R}(\boldsymbol{v})'_4 \leftarrow \mathcal{R}(\boldsymbol{v})'_4 - 1$ alters $\mathcal{R}(\boldsymbol{v})'$ into the $(3,1)$-read vector of $\boldsymbol{v} = (0,1,1,1,0,0)$ or $\boldsymbol{v} = (1,0,1,1,0,0)$ respectively.*

Henceforth, we represent the subsequence reconstructed using P7 from left to right with a noisy read sub-derivative, say $\Delta^\beta(\boldsymbol{x})'$, as $\hat{\boldsymbol{x}}^{(\beta)} = (\hat{x}_{\beta+1}, \hat{x}_{\beta+1+\ell}, \ldots, \hat{x}_{\beta+1+\lfloor \frac{n-\beta-1}{\ell} \rfloor \ell})$. Analogously, $\tilde{\boldsymbol{x}}^{(\beta)}$ corresponds to right to left reconstruction.

Lemma 6. *For $\ell \geq 3$, let $\mathcal{R}(\boldsymbol{x})'$ be a noisy $(\ell,1)$-read vector of $\boldsymbol{x} \in \Sigma_2^n$, such that for some $\alpha, \beta \in \Sigma_\ell$, where $\beta = (\alpha - 1) \bmod \ell$, $\mathrm{wt}(\Delta^\beta(\boldsymbol{x})') = -\mathrm{wt}(\Delta^\alpha(\boldsymbol{x})') \neq 0$. Reconstruction by P7 with $\Delta^\beta(\boldsymbol{x})'$ from left to right (respectively, right to left) yields $\hat{\boldsymbol{x}}^{(\beta)}$ ($\tilde{\boldsymbol{x}}^{(\beta)}$) for which we define i (j) as the minimum (maximum) index at which $\hat{x}_{\beta+i\ell+1} \notin \Sigma_2$ ($\tilde{x}_{\beta+j\ell+1} \notin \Sigma_2$), or $i = \lfloor \frac{n-\beta-1}{\ell} \rfloor + 1$ ($j = -1$) if no such index exists. Then, it holds that for all $j+2 \leq h \leq i-1$, $\mathcal{R}(\boldsymbol{x})'_{\beta+h\ell+1} = \mathcal{R}(\boldsymbol{x})'_{\beta+h\ell}$ and the error position in $\mathcal{R}(\boldsymbol{x})'$, say k, satisfies $\frac{k-\beta-1}{\ell} \in \{j+1, j+2, \ldots, i\}$.*

Example 5. *We reconsider $\mathcal{R}_{3,1}(\boldsymbol{v})'$ from Example 4. From $\Delta^0(\boldsymbol{v})' = (1,1,-1)$, we reconstruct $\hat{\boldsymbol{v}}^{(0)} = (1,2)$ and $\tilde{\boldsymbol{v}}^{(0)} = (0,1)$. Since $\hat{v}_4 \notin \Sigma_2$ and $\tilde{\boldsymbol{v}}^{(0)} \in \Sigma_2^2$, we set $i = 1$ and $j = -1$ as per Lemma 6. Thus, either $\mathcal{R}(\boldsymbol{x})'_1$ or $\mathcal{R}(\boldsymbol{v})'_4$ is noisy, implying that $\boldsymbol{v} = (1,0,1,1,0,0)$ or $\boldsymbol{v} = (0,1,1,1,0,0)$ respectively.*

Lemma 6 essentially suggests that attempting reconstruction with a noisy read sub-derivative may help to narrow down the error location even further. This finally allows us to arrive at

Theorem 5. *For $\ell \geq 3$, $\mathcal{C}(n,\ell)$ is a 1-substitution $(\ell,1)$-read code.*

Proof: Let $\mathcal{R}(\boldsymbol{x})'$ arise from a single substitution on $(\ell,1)$-read vector of some $\boldsymbol{x} \in \mathcal{C}(n,\ell)$. In light of Lemma 4, this proof is dedicated to errors of magnitude 1.

Upon identifying $\alpha, \beta \in \Sigma_\ell$ where $\beta = (\alpha - 1) \bmod \ell$, such that $\mathrm{wt}(\Delta^\beta(\boldsymbol{x})') = -\mathrm{wt}(\Delta^\alpha(\boldsymbol{x})') \neq 0$, we attempt reconstruction with $\Delta^\beta(\boldsymbol{x})'$ from left to right and from right to left to obtain $\hat{\boldsymbol{x}}^{(\beta)}$ and $\tilde{\boldsymbol{x}}^{(\beta)}$ respectively, and define indices i and j according to Lemma 6. Since for all $j+1 < h < i$, $\mathcal{R}(\boldsymbol{x})'_{\beta+h\ell+1} - \mathcal{R}(\boldsymbol{x})'_{\beta+h\ell} = 0$, and a run of 0s in $\Delta^\beta(x)'$ can be of length at most $2\log 2(n+\ell)+1$ due to the constraints imposed on $\mathcal{C}(n,\ell)$, we infer that $i - j - 2 \leq 2\log 2(n+\ell)+1$.

From Lemma 6, we know that the error exists somewhere in $(\mathcal{R}(\boldsymbol{x})'_{\beta+(j+1)\ell+1}, \mathcal{R}(\boldsymbol{x})'_{\beta+(j+2)\ell+1}, \ldots, \mathcal{R}(\boldsymbol{x})'_{\beta+i\ell+1})$, which is evidently a substring of $\mathcal{R}^\pi(\boldsymbol{x})'$ and has a length of at most $2\log 2(n+\ell)+3$. Since an error of magnitude 1 surely affects $\mathcal{R}^\pi(\boldsymbol{x})' \bmod 2$, which belongs to a code that corrects a substitution error localized to a window of $2\log 2(n+\ell)+4$ bits, we can uniquely recover $\mathcal{R}^\pi(\boldsymbol{x}) \bmod 2$, and by P4, also $\boldsymbol{x}$. ■

References

[1] D. Deamer, M. Akeson, and D. Branton, "Three decades of nanopore sequencing," *Nature Biotechnology*, vol. 34, no. 5, pp. 518–524, May 2016.

[2] A. H. Laszlo *et al.*, "Decoding long nanopore sequencing reads of natural DNA," *Nature Biotechnology*, vol. 32, no. 8, pp. 829–833, Aug. 2014.

[3] J. Kasianowicz, E. Brandin, D. Branton, and D. Deamer, "Characterization of individual polynucleotide molecules using a membrane channel," *Proc. Natl. Acad. Sci.*, vol. 93, no. 24, pp. 13 770–13 773, Nov. 1996.

[4] W. Mao, S. N. Diggavi, and S. Kannan, "Models and information-theoretic bounds for nanopore sequencing," *IEEE Trans. Inf. Theory*, vol. 64, no. 4, pp. 3216–3236, Apr. 2018.

[5] R. Hulett, S. Chandak, and M. Wootters, "On Coding for an Abstracted Nanopore Channel for DNA Storage," in *IEEE Intl. Symp. on Inf. Theory (ISIT)*, Melbourne, Australia, Jul. 2021, pp. 2465–2470.

[6] B. McBain, E. Viterbo, and J. Saunderson, "Finite-state semi-markov channels for nanopore sequencing," in *IEEE Intl. Symp. Inf. Theory (ISIT)*, Espoo, Finland, Jun. 2022, pp. 216–221.

[7] Y. M. Chee, A. Vardy, V. K. Vu, and E. Yaakobi, "Coding for transverse-reads in domain wall memories," in *IEEE Intl. Symp. Inf. Theory (ISIT)*, Melbourne, Australia, Jul. 2021, pp. 2924–2929.

[8] J. Chrisnata, H. M. Kiah, and E. Yaakobi, "Optimal reconstruction codes for deletion channels," in *IEEE Intl. Symp. Inf. Theory Appl. (ISITA)*, Kapolei, HI, USA, Oct. 2020, pp. 279–283.

[9] ——, "Correcting deletions with multiple reads," *IEEE Trans. Inf. Theory*, vol. 68, no. 11, pp. 7141–7158, Nov. 2022.

[10] D. E. Knuth, "The sandwich theorem," *The Electronic Journal of Combinatorics*, vol. 1, no. 1, A1, Apr. 1994.

[11] C. Schoeny, A. Wachter-Zeh, R. Gabrys, and E. Yaakobi, "Codes correcting a burst of deletions or insertions," *IEEE Trans. Inf. Theory*, vol. 63, no. 4, pp. 1971–1985, Apr. 2017.