Sequence Reconstruction for Limited-Magnitude Errors

Hengjia Wei and Moshe Schwartz, *Senior Member, IEEE*

***Abstract*— Motivated by applications to DNA storage, we study reconstruction and list-reconstruction schemes for integer vectors that suffer from limited-magnitude errors. We characterize the asymptotic size of the intersection of error balls in relation to the code's minimum distance. We also devise efficient reconstruction algorithms for various limited-magnitude error parameter ranges. We then extend these algorithms to the list-reconstruction scheme, and show the trade-off between the asymptotic list size and the number of required channel outputs. These results apply to all codes, without any assumptions on the code structure. Finally, we also study linear reconstruction codes with small intersection, as well as show a connection to list-reconstruction codes for the tandem-duplication channel.**

***Index Terms*— Reconstruction codes, list-reconstruction codes, limited-magnitude errors, integer codes.**

I. Introduction

THE *sequence-reconstruction problem*, which was first introduced by Levenshtein [17], considers a paradigm in which a sequence from some set is transmitted multiple times over a channel and the receiver needs to recover the transmitted sequence from the received sequences. It was originally motivated by the communication scenario where the only feasible strategy to combat errors is repeated transmission. Recently, it has been observed that this problem has a natural connection to DNA-based data storage systems. In such systems, the DNA strands are expected to be replicated many times, whether due to biological processes when in-vivo storage is used, or due to chemical processes in synthesis or sequencing when in-vitro storage is used. The user usually gets many noisy reads of the stored DNA strand when retrieving the data. Recovering the original DNA strand from its multiple noisy reads is therefore a sequence-reconstruction problem.

For a sequence $\mathbf{x}$, the *error ball* of $\mathbf{x}$ is the set of all possible outputs with $\mathbf{x}$ being transmitted through the channel. Clearly, the number of different channel outputs required to recover the transmitted sequence must be larger than the maximum intersection between the error balls of any two possible transmitted sequences [17]. One goal of the reconstruction problem is to determine the maximum intersection of two balls where the distance between their centers is at least some prescribed value. A significant number of papers has been devoted to determining this value for various error models, including substitutions, deletions, insertions, transpositions, and tandem-duplications [5], [12], [17], [18], [20], [31]. Additionally, a graph-theoretical approach was studied in [14], [17], and [15] to solve this problem in a more general metric distance.

Apart from determining the maximum intersection of two balls, other research directions have been considered as well. Refs. [1], [28] proposed efficient reconstruction algorithms to combat substitutions. Reconstruction codes have been designed to recover the transmitted sequences from a given number of sequences corrupted by tandem-duplications [31] or a single edit error [3]. Yaakobi and Bruck extended this problem in the context of associative memories [28] and introduced the notion of *uncertainty* of an associative memory for information retrieval, the value of which is equal to the maximum intersection of multiple error balls. In the context of sequence reconstruction, a closely related problem is to construct a list from multiple received noisy sequences such that the transmitted sequence is included in the list. The trade-off between the size of the minimum list (the number of balls) and the number of different received noisy sequences (the maximum intersection / uncertainty) has been analyzed for substitutions [10] and for tandem-duplications [32].

This paper focuses on *limited-magnitude errors*, which could be found in several applications, including high-density magnetic recording channels [13], [16], flash memories [4], and some DNA-based storage systems [9], [27]. In all of these applications, information is encoded as vectors of integers, and these vectors are affected by noise that may increase or decrease entries of the vectors by a limited amount. For instance, in a new inexpensive enzymatic method of DNA synthesis [9], the information is first encoded as sequences of the form $((a_1,u_1),(a_2,u_2),\dots,(a_n,u_n))$, where $a_i \in \{A,T,C,G\}$, $u_i \in \mathbb{Z}$ for $1 \leqslant i \leqslant n$, and $a_i \neq a_{i+1}$ for $1 \leqslant i \leqslant n-1$. Each pair (a_i,u_i) represents a run of a_i whose length is controlled[1] by u_i. In the molecule-synthesis process, the corresponding DNA strings which consist of n runs are synthesized. Due to variability, the length of each run might be shorter or longer than planned, usually by a limited amount.

Manuscript received August 22, 2021; revised February 27, 2022; accepted March 8, 2022. Date of publication March 14, 2022; date of current version June 15, 2022. This work was supported in part by the Israel Science Foundation (ISF) under Grant 270/18. *(Corresponding author: Hengjia Wei.)*

Hengjia Wei is with the Peng Cheng Laboratory, Shenzhen 518000, China (e-mail: hjwei05@gmail.com).

Moshe Schwartz is with the School of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Beer Sheva 8410501, Israel (e-mail: schwartz@ee.bgu.ac.il).

Communicated by A. Thangaraj, Associate Editor for Coding and Decoding.

Digital Object Identifier 10.1109/TIT.2022.3159736

[1]To combat minor variations, the set of positive integers is partitioned into several intervals, and the length of the i-th run could take any value from the interval identified by u_i.

The design of codes combating such errors, or equivalently, the packing/tiling of the corresponding errors balls, has been extensively researched, see e.g., [2], [7], [8], [11], [22]–[26], [29], [30], [33]–[35], and the many references therein.

In this paper, we study the reconstruction problem with respect to limited-magnitude errors. We first propose a new kind of distance to capture the capability of correcting limited-magnitude errors. Then for any code $\mathcal{C}$ of distance at least a prescribed value, we present both an upper bound and a lower bound on the size of the maximum intersection of any two error balls centered at the codewords of $\mathcal{C}$. In this way, we characterize the trade-off between this value and the number of excessive errors that the code $\mathcal{C}$ cannot cope with. Moreover, we study this reconstruction problem in a group-theoretical approach. In the channel that introduces a single limited-magnitude error, we design two classes of reconstruction codes which both have densities significantly larger than that of the normal error-correcting codes, at the cost of requiring one or two more received sequences. We present two efficient algorithms to reconstruct a transmitted codeword from any given code. Finally, we modify our reconstruction algorithms to accommodate the requirement of list decoding when the number of received sequences is less than the maximum intersection. Additionally, we show that one of our reconstruction algorithm could be used in the context of tandem duplications.

The paper is organized as follows. Section II provides notation and basic known results used throughout the paper. In Section III we study the maximum intersection and present a few upper bounds and lower bounds. In Section IV, we design codes that can recover the sequence from two or three received sequences. Section V presents two efficient reconstruction algorithms. Section VI studies the list decoding problem with multiple received sequences. Section VII discusses the reconstruction algorithm for tandem duplications.

II. Preliminaries

Let $\mathbb{Z}$ denote the ring of integers and $\mathbb{N}$ denote the set of natural numbers. Throughout the paper we let n and t be integers such that $n \geqslant t \geqslant 1$. We further assume k_+ and k_- are non-negative integers such that $k_+ \geqslant k_- \geqslant 0$. For integers $a \leqslant b$ we define $[a,b] \triangleq \{a, a+1, \dots, b\}$ and $[a,b]^* \triangleq [a,b] \setminus \{0\}$. Vectors will be written using bold lower-case letters. If $\mathbf{v} = (v_1, \dots, v_n)$ is a vector, we shall conveniently use $\mathbf{v}[i]$ to denote its i-th entry, namely, $\mathbf{v}[i] \triangleq v_i$.

For an integer vector $\mathbf{v} \in \mathbb{Z}^n$, if t of its entries suffer an increase by as much as k_+, or a decrease by as much as k_-, we say $\mathbf{v}$ suffers t (k_+, k_-)*-limited-magnitude errors*. We define the (n, t, k_+, k_-)*-error-ball* as

$$\mathcal{B}(n,t,k_+,k_-) \triangleq \{\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{Z}^n \mid -k_- \leqslant x_i \leqslant k_+ \text{ and } \mathrm{wt}(\mathbf{x}) \leqslant t\}, \quad (1)$$

where $\mathrm{wt}(\mathbf{x})$ denotes the Hamming weight of $\mathbf{x}$. Thus, the translate $\mathbf{v} + \mathcal{B}(n,t,k_+,k_-)$ is the error ball centered at the vector $\mathbf{v}$. If the values of k_+ and k_- can be inferred from the context, we simply denote it as $\mathcal{B}_t(\mathbf{v})$ to emphasize its center and radius.

The size of the ball $\mathcal{B}(n,t,k_+,k_-)$ will appear throughout this work. It is closely related to the size of a ball in the Hamming metric, which over an alphabet of size q, and with radius t, is

$$V_q(n,t) \triangleq \sum_{i=0}^{t} \binom{n}{i} (q-1)^i.$$

Using this notation, the size of $\mathcal{B}(n,t,k_+,k_-)$ is given by

$$|\mathcal{B}(n,t,k_+,k_-)| = V_{k_+ + k_- + 1}(n,t).$$

An error-correcting code in this setting is a packing of $\mathbb{Z}^n$ by $\mathcal{B}(n,t,k_+,k_-)$, that is, a subset $\mathcal{C} \subseteq \mathbb{Z}^n$ such that for any two distinct vectors $\mathbf{x}, \mathbf{y} \in \mathcal{C}$, the balls $\mathbf{x} + \mathcal{B}(n,t,k_+,k_-)$ and $\mathbf{y} + \mathcal{B}(n,t,k_+,k_-)$ are disjoint. The largest integer t with this property is the *unique-decoding radius of* $\mathcal{C}$ (with respect to (k_+, k_-)-limited-magnitude errors).

Throughout this paper, we use these two notions of error-correcting code and packing interchangeably. The following distance,[2] d_ℓ, allows to determine the number of $(k_+, 0)$-limited-magnitude errors that a code could correct.

Definition 1 ([4]): For $\mathbf{x}, \mathbf{y} \in \mathbb{Z}^n$, define

$$N(\mathbf{x}, \mathbf{y}) \triangleq |\{i; \mathbf{x}[i] > \mathbf{y}[i]\}|.$$

The distance d_ℓ between $\mathbf{x}$ and $\mathbf{y}$ is defined as

$$d_\ell(\mathbf{x},\mathbf{y}) \triangleq \begin{cases} n+1, & \max_i\{|\mathbf{x}[i] - \mathbf{y}[i]|\} > \ell, \\ \max\{N(\mathbf{x},\mathbf{y}), N(\mathbf{y},\mathbf{x})\}, & \text{otherwise.} \end{cases}$$

Proposition 2 ([4]): A code $\mathcal{C} \in \mathbb{Z}^n$ can correct e $(k_+, 0)$-limited-magnitude errors if and only if $d_{k_+}(\mathbf{x}, \mathbf{y}) \geqslant e + 1$ for all distinct $\mathbf{x}, \mathbf{y} \in \mathcal{C}$.

If the number of errors, t, exceeds the unique-decoding radius, e, of $\mathcal{C}$, the error balls of radius t centered at the codewords might intersect. For any two distinct vectors $\mathbf{x}, \mathbf{y} \in \mathbb{Z}^n$, let $N(\mathbf{x}, \mathbf{y}; t, k_+, k_-)$ be the size of the intersection of the two balls $\mathbf{x} + \mathcal{B}(n,t,k_+,k_-)$ and $\mathbf{y} + \mathcal{B}(n,t,k_+,k_-)$, i.e.,

$$N(\mathbf{x}, \mathbf{y}; t, k_+, k_-) \triangleq |(\mathbf{x} + \mathcal{B}(n,t,k_+,k_-)) \cap (\mathbf{y} + \mathcal{B}(n,t,k_+,k_-))|.$$

Given a code $\mathcal{C} \subseteq \mathbb{Z}^n$, let $N(\mathcal{C}; t, k_+, k_-)$ be the size of the maximum intersection of any two balls centered at different codewords of $\mathcal{C}$, that is,

$$\begin{aligned} N(\mathcal{C}; t, k_+, k_-) &\triangleq \max_{\substack{\mathbf{x},\mathbf{y} \in \mathcal{C} \\ \mathbf{x} \neq \mathbf{y}}} \{N(\mathbf{x}, \mathbf{y}; t, k_+, k_-)\} \\ &= \max_{\substack{\mathbf{x},\mathbf{y} \in \mathcal{C} \\ \mathbf{x} \neq \mathbf{y}}} \{|(\mathbf{x} + \mathcal{B}(n,t,k_+,k_-)) \cap (\mathbf{y} + \mathcal{B}(n,t,k_+,k_-))|\}. \end{aligned}$$

In Section III we shall first extend the definition of d_ℓ to capture the error-correcting capability for $k_- > 0$, then give some upper bounds and lower bounds on $N(\mathcal{C}; t, k_+, k_-)$ for any code $\mathcal{C}$ of distance at least a prescribed value. In this way, we show that for any fixed integers t and e with $t > e$, a code $\mathcal{C}$ can correct up to e (k_+, k_-)-limited-magnitude errors if

[2] Note that although d_ℓ is referred to as a distance in [4] and here, it does not satisfy the triangle inequality.

and only if the maximum intersection $N(\mathcal{C};t,k_+,k_-)$ has size $\Theta(n^{t-e-1})$.

A. Lattice Code/Packing and Group Splitting

Let G be a finite Abelian group, where $+$ denotes the group operation. For $m\in\mathbb{Z}$ and $g\in G$, let mg denote $g+g+\cdots+g$ (with m copies of g) when $m>0$, which is extended in the natural way to $m\leqslant 0$, i.e., the sum of $|m|$ copies of $-g$ (the additive inverse of g in G).

A *lattice* is an additive subgroup Λ of $\mathbb{Z}^n$. Throughout the paper we shall assume lattices are non-degenerate, namely, the quotient group $\mathbb{Z}^n/\Lambda$ is a finite group. The *density* of Λ is defined as $|\mathbb{Z}^n/\Lambda|^{-1}$. Let G be an Abelian group and $\mathbf{s}=(s_1,s_2,\ldots,s_n)$ be a sequence of G^n. Define

$$\Lambda\triangleq\{\mathbf{x}\in\mathbb{Z}^n;\mathbf{x}\cdot\mathbf{s}=0\}.$$

Then Λ is a lattice. Conversely, every lattice $\Lambda\subseteq\mathbb{Z}^n$ can be represented in this form with some sequence $\mathbf{s}$ over some Abelian group G: Let G be the quotient group, i.e., $G=\mathbb{Z}^n/\Lambda$. Let $\phi:\mathbb{Z}^n\to G$ be the natural homomorphism, namely the one that maps any $\mathbf{x}\in\mathbb{Z}^n$ to the coset of Λ in which it resides. Let $\mathbf{e}_i$ be the i-th unit vector in $\mathbb{Z}^n$ and set $s_i\triangleq\phi(\mathbf{e}_i)$ for all $1\leqslant i\leqslant n$ and $\mathbf{s}\triangleq(s_1,s_2,\ldots,s_n)$. Then $\mathbf{s}$ is a vector over the Abelian group G and $\Lambda=\ker\phi=\{\mathbf{x}\in\mathbb{Z}^n;\mathbf{x}\cdot\mathbf{s}=0\}$. Note that if we treat Λ as a code, the vector $\mathbf{s}$ plays the role of a "parity-check matrix".

A lattice code correcting t (k_+,k_-)-limited magnitude errors is equivalent to a lattice packing of $\mathbb{Z}^n$ with $\mathcal{B}(n,t,k_+,k_-)$. Their connection to group splitting for $t=1$ has been observed in [21]. For an excellent treatment and history, the reader is referred to [25] and the many references therein. Recently, an extended definition of group splitting was proposed [26] in connection with lattice packings of $\mathcal{B}(n,t,k_+,k_-)$ with $t>1$, i.e., lattice codes that correct multiple errors.

Definition 3 ([26]): Let G be a finite Abelian group. Let $M\subseteq\mathbb{Z}\setminus\{0\}$ be a finite set, and $\mathbf{s}=(s_1,s_2,\ldots,s_n)\in G^n$. If the elements $\mathbf{e}\cdot\mathbf{s}$, where $\mathbf{e}\in(M\cup\{0\})^n$ and $1\leqslant\mathrm{wt}(\mathbf{e})\leqslant t$, are all distinct and non-zero in G, we say the set M partially t-splits G with a splitter vector $\mathbf{s}$, denoted

$$G\geqslant M\diamond_t\mathbf{s}.$$

In our context of (k_+,k_-)-limited-magnitude errors, we need to take $M\triangleq[-k_-,k_+]^*$. The following theorem shows the equivalence of partial t-splittings with M and lattice packings of $\mathcal{B}(n,t,k_+,k_-)$.

Theorem 4 ([26]): Let $\Lambda\subseteq\mathbb{Z}^n$ be a lattice. Let $G=\mathbb{Z}^n/\Lambda$. Set $\mathbf{s}\triangleq(\phi(\mathbf{e}_1),\phi(\mathbf{e}_2),\ldots,\phi(\mathbf{e}_n))$, where $\phi:\mathbb{Z}^n\to G$ is the natural homomorphism. Then $\mathcal{B}(n,t,k_+,k_-)$ lattice packs $\mathbb{Z}^n$ by Λ if and only if $G\geqslant[-k_-,k_+]^*\diamond_t\mathbf{s}$.

Recall that the sequence $\mathbf{s}$ plays the role of "parity-check matrix". Then the elements $\mathbf{e}\cdot\mathbf{s}$, where $\mathbf{e}\in(M\cup\{0\})^n$ and $1\leqslant\mathrm{wt}(\mathbf{e})\leqslant t$, correspond to the syndromes. Theorem 4 tells us that the code Λ can correct limited-magnitude errors if and only if the syndromes are distinct.

Example 5: Consider the ball $\mathcal{B}(2,1,3,2)$. Let Λ be a lattice of $\mathbb{Z}^2$ generated by $(4,1)$ and $(3,5)$. Then $\mathbb{Z}^n/\Lambda\cong\mathbb{Z}_{17}$ and $\Lambda=\{\mathbf{x}\in\mathbb{Z}^2;\mathbf{x}\cdot\mathbf{s}=0\}$ with $\mathbf{s}=(4,1)$. The ball size is $|\mathcal{B}(2,1,3,2)|=11$, comprising of the no-error case, and ten distinct error scenarios. The ten syndromes are $1,2,3,4,8,9,12,13,15,16$, which are pairwise distinct. Thus, the ball $\mathcal{B}(2,1,3,2)$ lattice packs $\mathbb{Z}^2$ by Λ.

III. Maximum Intersection of Two Error Balls

In this section we study the size of the maximum intersection $N(\mathcal{C};t,k_+,k_-)$ for any given code $\mathcal{C}\subseteq\mathbb{Z}^n$. This is an essential component in analyzing reconstruction codes since it determines the number of distinct channel outputs needed for the reconstruction to be successful. We first look at the case of $\mathcal{C}=\mathbb{Z}^n$.

Theorem 6: For any n,t,k_+,k_- with $t\leqslant n$ and $0\leqslant k_-\leqslant k_+$, we have that

$$\begin{aligned}N(\mathbb{Z}^n;t,k_+,k_-)&=\sum_{i=0}^{t-1}\binom{n-1}{i}(k_++k_-)^{i+1}\\&=(k_++k_-)V_{k_++k_-+1}(n-1,t-1).\end{aligned}$$

Proof: Consider the two words $\mathbf{x}=(0,0,0,\ldots,0)$ and $\mathbf{y}=(1,0,0,\ldots,0)$ in $\mathbb{Z}^n$. Then it is easy to see that the intersection of the two balls centered at $\mathbf{x}$ and $\mathbf{y}$ has size $(k_++k_-)\sum_{i=0}^{t-1}\binom{n-1}{i}(k_++k_-)^i$.

Denote $N\triangleq(k_++k_-)\sum_{i=0}^{t-1}\binom{n-1}{i}(k_++k_-)^i$. In the following, we shall show that $N(\mathbb{Z}^n;t,k_+,k_-)\leqslant N$ by giving a decoding algorithm based on the majority rule. Fix an arbitrary vector $\mathbf{x}\in\mathbb{Z}^n$. Suppose that $\mathbf{z}_1,\mathbf{z}_2,\ldots,\mathbf{z}_{N+1}\in\mathbf{x}+\mathcal{B}(n,t,k_+,k_-)$. For each $i\in[1,n]$, define the *multiset*

$$\mathcal{Z}_i\triangleq\{\mathbf{z}_1[i],\mathbf{z}_2[i],\ldots,\mathbf{z}_{N+1}[i]\}.$$

Let m_i be the smallest element of $\mathcal{Z}_i$ and M_i be the largest element of $\mathcal{Z}_i$. If $M_i-m_i=k_++k_-$, necessarily $\mathbf{x}[i]=m_i+k_-$. Otherwise, $M_i-m_i<k_++k_-$ and there are at most k_++k_- distinct elements in $\mathcal{Z}_i$. For each $m_i\leqslant a\leqslant M_i$, if $a\neq\mathbf{x}[i]$, then the number of $\mathbf{z}_j$'s such that $\mathbf{z}_j[i]=a$ is at most $\sum_{i=0}^{t-1}\binom{n-1}{i}(k_++k_-)^i$. Since $N+1=(k_++k_-)\sum_{i=0}^{t-1}\binom{n-1}{i}(k_++k_-)^i+1$, $\mathbf{x}[i]$ must be the most frequently occurring element of $\mathcal{Z}_i$. ■

We now move on to study the case where $\mathcal{C}\subseteq\mathbb{Z}^n$ is a code of distance at least a prescribed value. We first consider the case of $k_-=0$.

Lemma 7: Let $\delta\leqslant t\leqslant n$. For any two vectors $\mathbf{x},\mathbf{y}\in\mathbb{Z}^n$ with $d_{k_+}(\mathbf{x},\mathbf{y})=\delta\leqslant n$, we have that

$$\begin{aligned}&\sum_{i=0}^{t-\delta}\binom{n-2\delta}{i}(k_+)^i\leqslant N(\mathbf{x},\mathbf{y};t,k_+,0)\\&\quad\leqslant\sum_{i=0}^{t-\delta}\binom{n-\delta}{i}(k_+)^i\sum_{k=\delta+i-t}^{t-i}\binom{\delta}{k}(k_+-1)^{\delta-k}.\end{aligned}\qquad(2)$$

Proof: Assume that $N(\mathbf{x},\mathbf{y})=\delta$ and $N(\mathbf{y},\mathbf{x})=\delta'$ with $\delta'\leqslant\delta$, where $N(\mathbf{x},\mathbf{y})\triangleq|\{i;\mathbf{x}[i]>\mathbf{y}[i]\}|$. Let $\mathbf{z}$ be an element in $\mathcal{B}_t(\mathbf{x})\cap\mathcal{B}_t(\mathbf{y})$. Denote by i the number of positions where $\mathbf{x}$ and $\mathbf{y}$ agree but differ from $\mathbf{z}$. Denote by j the number of positions where $\mathbf{x}$ and $\mathbf{z}$ agree but differ from $\mathbf{y}$. Since

$k_- = 0$, in these positions the components of $\mathbf{x}$ must be larger than those of $\mathbf{y}$. Thus,

$$0 \leqslant j \leqslant \delta. \tag{3}$$

Denote by k the number of positions where $\mathbf{y}$ and $\mathbf{z}$ agree but differ from $\mathbf{x}$. In these positions the components of $\mathbf{y}$ must be larger than those of $\mathbf{x}$, and so,

$$0 \leqslant k \leqslant \delta'. \tag{4}$$

There are $(\delta - j) + \delta' + i$ positions where $\mathbf{x}$ and $\mathbf{z}$ differ and $\delta + (\delta' - k) + i$ positions where $\mathbf{y}$ and $\mathbf{z}$ differ. Hence,

$$0 \leqslant \delta - j + \delta' + i \leqslant t, \tag{5}$$
$$0 \leqslant \delta + \delta' - k + i \leqslant t. \tag{6}$$

Combine (4) and (6) to get

$$0 \leqslant i \leqslant t - \delta.$$

Combine (3)–(6) to get

$$\delta + \delta' + i - t \leqslant j \leqslant t - i,$$

where the first inequality comes from (5) and the second inequality is obtained by combining (3), (4) and (6). Similarly, we have

$$\delta + \delta' + i - t \leqslant k \leqslant t - i.$$

Hence, the number of choices for $\mathbf{z}$ is at least

$$\sum_{i=0}^{t-\delta} \binom{n-\delta-\delta'}{i} (k_+)^i;$$

and at most

$$\sum_{i=0}^{t-\delta} \binom{n-\delta-\delta'}{i} (k_+)^i \sum_{j=\delta+\delta'+i-t}^{t-i} \binom{\delta}{j} \times \sum_{k=\delta+\delta'+i-t}^{t-i} \binom{\delta'}{k} (k_+ - 1)^{\delta+\delta'-j-k}. \tag{7}$$

Thus, we just proved the lower bound, since $\binom{n-\delta-\delta'}{i} \geqslant \binom{n-2\delta}{i}$. In the following, we show that (7) is decreasing with δ'. Note that (7) is achieved only when $|\mathbf{x}[i] - \mathbf{y}[i]| \leqslant 1$ for all $1 \leqslant i \leqslant n$. W.l.o.g., we assume that

$$\mathbf{x} = (\underbrace{1,1,\dots,1}_{\delta}, \underbrace{0,0,\dots,0}_{\delta'}, \underbrace{0,0,\dots,0}_{n-\delta-\delta'}),$$

and

$$\mathbf{y} = (\underbrace{0,0,\dots,0}_{\delta}, \underbrace{1,1,\dots,1}_{\delta'}, \underbrace{0,0,\dots,0}_{n-\delta-\delta'}).$$

Let

$$\mathbf{y}' = (\underbrace{0,0,\dots,0}_{\delta}, \underbrace{1,1,\dots,1}_{\delta'-1}, \underbrace{0,0,\dots,0}_{n-\delta-\delta'+1}).$$

We are going to show that

$$|\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y})| \leqslant |\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y}')|. \tag{8}$$

For any $\mathbf{z} \in (\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y})) \setminus (\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y}'))$, we have that $\mathbf{z}[\delta + \delta'] \in [1, k_+]$. Furthermore, since $\mathbf{z} \in \mathcal{B}_t(\mathbf{y})$ and $\mathbf{z} \notin \mathcal{B}_t(\mathbf{y}')$, necessarily $\mathbf{z}[\delta + \delta'] = 1$.

Let $\mathbf{z}'$ be the vector obtained from $\mathbf{z}$ by changing $\mathbf{z}[\delta + \delta']$ from '1' to '0'. Then it is easy to verify that $\mathbf{z}' \in \mathcal{B}_t(\mathbf{x})$, $\mathbf{z}' \in \mathcal{B}_t(\mathbf{y}')$ and $\mathbf{z}' \notin \mathcal{B}_t(\mathbf{y})$. Hence,

$$\mathbf{z}' \in (\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y}')) \setminus (\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y})).$$

Note that for different choices of $\mathbf{z}$, $\mathbf{z}'$ are pairwise distinct. Therefore, we have proved (8), and so (7) is decreasing with δ'. The upper bound in (2) follows from (7) by taking $\delta' = 0$. ■

Remark: The lower bound in (2) can be attained if $N(\mathbf{x}, \mathbf{y}) = N(\mathbf{y}, \mathbf{x}) = \delta$ and $\mathbf{x}[i] - \mathbf{y}[i] \in \{k_+, 0, -k_+\}$ for all $1 \leqslant i \leqslant n$; the upper bound in (2) can be attained if $N(\mathbf{y}, \mathbf{x}) = 0$ and $\mathbf{x}[i] - \mathbf{y}[i] \in \{0, 1\}$ for all $1 \leqslant i \leqslant n$.

With Lemma 7 in hand, we can bound the intersection of balls around codewords in a general code, and with $k_- = 0$.

Lemma 8: Let $\delta \leqslant t \leqslant n$, and $\mathcal{C} \subseteq \mathbb{Z}^n$ be a code with minimum distance $d_{k_+}(\mathcal{C}) = \delta$. Then

$$\sum_{i=0}^{t-\delta} \binom{n-2\delta}{i} (k_+)^i \leqslant N(\mathcal{C}; t, k_+, 0) \leqslant \sum_{i=0}^{t-\delta} \binom{n-\delta}{i} (k_+)^i \sum_{k=\delta+i-t}^{t-i} \binom{\delta}{k} (k_+ - 1)^{\delta-k}.$$

Proof: It suffices to show that the upper bound in (2) is decreasing with δ. The proof is the same as that in the proof of Lemma 7. According to the remark after Lemma 7, w.l.o.g, we assume that

$$\mathbf{x} = (\underbrace{1,1,\dots,1}_{\delta}, \underbrace{0,0,\dots,0}_{n-\delta}),$$

and

$$\mathbf{y} = (\underbrace{0,0,\dots,0}_{\delta}, \underbrace{0,0,\dots,0}_{n-\delta}).$$

Let

$$\mathbf{x}' = (\underbrace{1,1,\dots,1}_{\delta-1}, \underbrace{0,0,\dots,0}_{n-\delta+1}).$$

For any $\mathbf{z} \in (\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y})) \setminus (\mathcal{B}_t(\mathbf{x}') \cap \mathcal{B}_t(\mathbf{y}))$, we have that $\mathbf{z}[\delta] = 1$. Let $\mathbf{z}'$ be the vector obtained from $\mathbf{z}$ by changing $\mathbf{z}[\delta]$ from '1' to '0'. Then $\mathbf{z}' \in (\mathcal{B}_t(\mathbf{x}') \cap \mathcal{B}_t(\mathbf{y})) \setminus (\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y}))$. Hence, $|\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y})| \leqslant |\mathcal{B}_t(\mathbf{x}') \cap \mathcal{B}_t(\mathbf{y})|$. ■

Theorem 9: Let δ, t and k_+ be fixed integers such that $1 \leqslant \delta \leqslant t \leqslant n$. For any code $\mathcal{C} \subseteq \mathbb{Z}^n$, $N(\mathcal{C}; t, k_+, 0) = \Theta(n^{t-\delta})$ if and only of $\mathcal{C}$ can correct up to $\delta - 1$ $(k_+, 0)$-limited-magnitude errors.

Proof: According to Theorem 6 and Lemma 8, $N(\mathcal{C}; t, k_+, 0) = \Theta(n^{t-\delta})$ if and only if $d_{k_+}(\mathcal{C}) = \delta$. Combining this with Proposition 2, the theorem is proved. ■

Now, we direct our attention to the case of $k_- > 0$. We first extend the distance d_{k_+} from Definition 1.

Definition 10: For $\mathbf{x},\mathbf{y}\in\mathbb{Z}^n$, we define

$$\begin{aligned}N_{k_-}(\mathbf{x},\mathbf{y}) &\triangleq \left|\{i; 0<|\mathbf{x}[i]-\mathbf{y}[i]|\leqslant k_-\}\right|,\\ N_{k_+,k_-}(\mathbf{x},\mathbf{y}) &\triangleq \left|\{i; k_+<|\mathbf{x}[i]-\mathbf{y}[i]|\leqslant k_++k_-\}\right|,\\ M_{k_+,k_-}(\mathbf{x},\mathbf{y}) &\triangleq \left|\{i; k_-<\mathbf{x}[i]-\mathbf{y}[i]\leqslant k_+\}\right|,\end{aligned}$$

where we note that $M_{k_+,k_-}(\mathbf{x},\mathbf{y})$ and $M_{k_+,k_-}(\mathbf{y},\mathbf{x})$ are not necessarily the same.

If $\max_i\{|\mathbf{x}[i]-\mathbf{y}[i]|\}>k_++k_-$, define the distance d_{k_+,k_-} between $\mathbf{x}$ and $\mathbf{y}$ to be $n+1$; otherwise, the distance d_{k_+,k_-} is defined as

$$\begin{aligned}&d_{k_+,k_-}(\mathbf{x},\mathbf{y})\\ &\triangleq \left\lceil \frac{1}{2}\max(N_{k_-}(\mathbf{x},\mathbf{y}) - \left|M_{k_+,k_-}(\mathbf{x},\mathbf{y})-M_{k_+,k_-}(\mathbf{y},\mathbf{x})\right|,0)\right\rceil\\ &\quad+\max\left(M_{k_+,k_-}(\mathbf{x},\mathbf{y}),M_{k_+,k_-}(\mathbf{y},\mathbf{x})\right)+N_{k_+,k_-}(\mathbf{x},\mathbf{y}).\end{aligned}$$

It is worth noting that when $k_-=0$, the distance $d_{k_+,0}$ defined above coincides with the distance d_{k_+} in Definition 1.

Proposition 11: A code $\mathcal{C}\subseteq\mathbb{Z}^n$ can correct t (k_+,k_-)-limited-magnitude errors if and only if $d_{k_+,k_-}(\mathbf{x},\mathbf{y})\geqslant t+1$ for all distinct $\mathbf{x},\mathbf{y}\in\mathcal{C}$.

Proof: ($\Leftarrow$) Let $\mathbf{x},\mathbf{y}\in\mathcal{C}$ be two codewords, and let $\mathbf{e},\mathbf{e}'\in\mathcal{B}(n,t,k_+,k_-)$ be two error vectors, such that $\mathbf{x}\neq\mathbf{y}$ or $\mathbf{e}\neq\mathbf{e}'$. Assume to the contrary that $\mathbf{x}+\mathbf{e}=\mathbf{y}+\mathbf{e}'$. Then $\mathbf{x}-\mathbf{y}=\mathbf{e}'-\mathbf{e}$, and so $d_{k_+,k_-}(\mathbf{x},\mathbf{y})=d_{k_+,k_-}(\mathbf{e},\mathbf{e}')$. Let $n_1=N_{k_-}(\mathbf{e},\mathbf{e}')$, $n_2=N_{k_+,k_-}(\mathbf{e},\mathbf{e}')$, $m_1=M_{k_+,k_-}(\mathbf{e},\mathbf{e}')$, and $m_2=M_{k_+,k_-}(\mathbf{e}',\mathbf{e})$. W.l.o.g., assume that $m_1\geqslant m_2$. Since both $\mathbf{e},\mathbf{e}'$ have Hamming weight at most t and $\mathbf{e},\mathbf{e}'\in[-k_-,k_+]^n$, then

$$m_1+n_2\leqslant t.$$

Next, let $\mathcal{N}_1\triangleq\{i;0<|\mathbf{e}[i]-\mathbf{e}'[i]|\leqslant k_-\}$. Consider the two subsets

$$P_1\triangleq\{i;\mathbf{e}[i]\neq 0\}\cap\mathcal{N}_1,$$

$$P_2\triangleq\{i;\mathbf{e}'[i]\neq 0\}\cap\mathcal{N}_1.$$

Since $\mathbf{e}[i]\neq\mathbf{e}'[i]$ for each $i\in\mathcal{N}_1$, necessarily $\mathcal{N}_1=P_1\cup P_2$. Furthermore, we have $|P_1|+m_1+n_2\leqslant t$ and $|P_2|+m_2+n_2\leqslant t$, as $\mathrm{wt}(\mathbf{e}),\mathrm{wt}(\mathbf{e}')\leqslant t$. Thus,

$$\begin{aligned}&n_1+(m_1+n_2)+(m_2+n_2)\\ &\qquad\leqslant|P_1|+|P_2|+(m_1+n_2)+(m_2+n_2)\leqslant 2t.\end{aligned}$$

Hence, $d_{k_+,k_-}(\mathbf{e},\mathbf{e}')=\lceil\max\{n_1-m_1+m_2,0\}/2\rceil+m_1+n_2\leqslant t$, which contradicts that $d_{k_+,k_-}(\mathbf{x},\mathbf{y})\geqslant t+1$.

($\Rightarrow$) Suppose that there are two distinct codewords $\mathbf{x},\mathbf{y}\in\mathcal{C}$ such that $d_{k_+,k_-}(\mathbf{x},\mathbf{y})\leqslant t$. Since d_{k_+,k_-} is symmetric, assume w.l.o.g. that $M_{k_+,k_-}(\mathbf{x},\mathbf{y})\leqslant M_{k_+,k_-}(\mathbf{y},\mathbf{x})$. Denote,

$$\begin{aligned}\mathcal{N}_1&\triangleq\{i;0<|\mathbf{x}[i]-\mathbf{y}[i]|\leqslant k_-\},\\ \mathcal{N}_2&\triangleq\{i;k_+<|\mathbf{x}[i]-\mathbf{y}[i]|\leqslant k_++k_-\},\\ \mathcal{M}_1&\triangleq\{i;k_-<\mathbf{x}[i]-\mathbf{y}[i]\leqslant k_+\},\\ \mathcal{M}_2&\triangleq\{i;k_-<\mathbf{y}[i]-\mathbf{x}[i]\leqslant k_+\}.\end{aligned}$$

Take an arbitrary subset $\mathcal{N}_1'\subseteq\mathcal{N}_1$ of size $\lceil\frac{1}{2}\max\{N_{k_-}(\mathbf{x},\mathbf{y})+M_{k_+,k_-}(\mathbf{x},\mathbf{y})-M_{k_+,k_-}(\mathbf{y},\mathbf{x}),0\}\rceil$. Let $\mathbf{e}$ be the vector with support set $\mathcal{N}_1'\cup\mathcal{N}_2\cup\mathcal{M}_2$, where $\mathbf{e}[i]=\mathbf{y}[i]-\mathbf{x}[i]$ when $i\in\mathcal{N}_1'\cup\mathcal{M}_2$, $\mathbf{e}[i]=\mathbf{y}[i]-\mathbf{x}[i]-k_-$ when $\mathbf{y}[i]-\mathbf{x}[i]>k_+$ and $\mathbf{e}[i]=\mathbf{y}[i]-\mathbf{x}[i]+k_+$ when $\mathbf{y}[i]-\mathbf{x}[i]<-k_+$. Let $\mathbf{e}'$ be the vector with support set $(\mathcal{N}_1\setminus\mathcal{N}_1')\cup\mathcal{N}_2\cup\mathcal{M}_1$, where $\mathbf{e}'[i]=\mathbf{x}[i]-\mathbf{y}[i]$ when $i\in(\mathcal{N}_1\setminus\mathcal{N}_1')\cup\mathcal{M}_1$, $\mathbf{e}'[i]=-k_-$ when $\mathbf{y}[i]-\mathbf{x}[i]>k_+$ and $\mathbf{e}'[i]=k_+$ when $\mathbf{y}[i]-\mathbf{x}[i]<-k_+$.

It follows that $\mathbf{e}'=\mathbf{x}-\mathbf{y}+\mathbf{e}$, i.e., $\mathbf{x}+\mathbf{e}=\mathbf{y}+\mathbf{e}'$, and $\mathbf{e},\mathbf{e}'\in[-k_-,k_+]^n$. In the following, we verify that both $\mathbf{e}$ and $\mathbf{e}'$ have Hamming weight at most t, which contradicts the fact that $\mathcal{C}$ can correct t (k_+,k_-)-limited-magnitude errors. The vector $\mathbf{e}$ has Hamming weight

$$\begin{aligned}&|\mathcal{N}_1'\cup\mathcal{M}_2\cup\mathcal{N}_2|\\ &=\left\lceil\frac{1}{2}\max(N_{k_-}(\mathbf{x},\mathbf{y})+M_{k_+,k_-}(\mathbf{x},\mathbf{y})-M_{k_+,k_-}(\mathbf{y},\mathbf{x}),0)\right\rceil\\ &\quad+M_{k_+,k_-}(\mathbf{y},\mathbf{x})+N_{k_+,k_-}(\mathbf{x},\mathbf{y})\\ &=\ d_{k_+,k_-}(\mathbf{x},\mathbf{y})\leqslant t.\end{aligned}$$

For the vector $\mathbf{e}'$, if $N_{k_-}(\mathbf{x},\mathbf{y})+M_{k_+,k_-}(\mathbf{x},\mathbf{y})-M_{k_+,k_-}(\mathbf{y},\mathbf{x})\leqslant 0$, then it has Hamming weight

$$\begin{aligned}&|(\mathcal{N}_1\setminus\mathcal{N}_1')\cup\mathcal{M}_1\cup\mathcal{N}_2|\\ &\leqslant N_{k_-}(\mathbf{x},\mathbf{y})+M_{k_+,k_-}(\mathbf{x},\mathbf{y})+N_{k_+,k_-}(\mathbf{x},\mathbf{y})\\ &\leqslant M_{k_+,k_-}(\mathbf{y},\mathbf{x})+N_{k_+,k_-}(\mathbf{x},\mathbf{y})\\ &=d_{k_+,k_-}(\mathbf{x},\mathbf{y})\leqslant t.\end{aligned}$$

Otherwise, it has Hamming weight

$$\begin{aligned}&|(\mathcal{N}_1\setminus\mathcal{N}_1')\cup\mathcal{M}_1\cup\mathcal{N}_2|\\ &\quad\leqslant\frac{1}{2}\left(N_{k_-}(\mathbf{x},\mathbf{y})-M_{k_+,k_-}(\mathbf{x},\mathbf{y})+M_{k_+,k_-}(\mathbf{y},\mathbf{x})\right)\\ &\qquad+M_{k_+,k_-}(\mathbf{x},\mathbf{y})+N_{k_+,k_-}(\mathbf{x},\mathbf{y})\\ &\quad=\frac{1}{2}\left(N_{k_-}(\mathbf{x},\mathbf{y})+M_{k_+,k_-}(\mathbf{x},\mathbf{y})-M_{k_+,k_-}(\mathbf{y},\mathbf{x})\right)\\ &\qquad+M_{k_+,k_-}(\mathbf{y},\mathbf{x})+N_{k_+,k_-}(\mathbf{x},\mathbf{y})\\ &\quad\leqslant d_{k_+,k_-}(\mathbf{x},\mathbf{y})\leqslant t.\end{aligned}$$

■

Lemma 12: Let $\mathbf{x}=(x_1,x_2,\ldots,x_n)$ and $\mathbf{y}=(y_1,y_2,\ldots,y_n)$ be two vectors of $\mathbb{Z}^n$. Denote $\mathbf{x}'\triangleq(y_1,x_2,\ldots,x_n)$ and $\mathbf{y}'\triangleq(x_1,y_2,\ldots,y_n)$. If $|x_1-y_1|\leqslant k_-$ or $|x_1-y_1|>k_+$, then

$$|\mathcal{B}_t(\mathbf{x})\cap\mathcal{B}_t(\mathbf{y})|=|\mathcal{B}_t(\mathbf{x}')\cap\mathcal{B}_t(\mathbf{y}')|.$$

Proof: We shall show that the size of $\mathcal{B}_t(\mathbf{x})\cap\mathcal{B}_t(\mathbf{y})$ is no more than that of $\mathcal{B}_t(\mathbf{x}')\cap\mathcal{B}_t(\mathbf{y}')$. Then the equality holds by switching $\{\mathbf{x},\mathbf{y}\}$ and $\{\mathbf{x}',\mathbf{y}'\}$. Let $\mathbf{z}=(z_1,z_2,\ldots,z_n)$ be an arbitrary vector of $\mathcal{B}_t(\mathbf{x})\cap\mathcal{B}_t(\mathbf{y})$. Denote

$$\mathbf{z}'\triangleq\begin{cases}\mathbf{z} & \text{if } z_1\notin\{x_1,y_1\},\\ (y_1,z_2,z_3,\ldots,z_n) & \text{if } z_1=x_1,\\ (x_1,z_2,z_3,\ldots,z_n) & \text{if } z_1=y_1.\end{cases}$$

If $z_1\notin\{x_1,y_1\}$, it is easy to see that $\mathbf{z}'\in\mathcal{B}_t(\mathbf{x}')\cap\mathcal{B}_t(\mathbf{y}')$. If $|x_1-y_1|>k_+$, we cannot have $z_1\in\{x_1,y_1\}$. However, if $z_1\in\{x_1,y_1\}$, necessarily $|x_1-y_1|\leqslant k_+$, and so, according to the lemma's condition we have $|x_1 - y_1| \leqslant k_-$. Then it is verifiable again that $\mathbf{z}'$ belongs to $\mathcal{B}_t(\mathbf{x}') \cap \mathcal{B}_t(\mathbf{y}')$. Noting that the map that sends $\mathbf{z}$ to $\mathbf{z}'$ is injective, we get $|\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y})| \leqslant |\mathcal{B}_t(\mathbf{x}') \cap \mathcal{B}_t(\mathbf{y}')|$. ■

Lemma 13: Let $\mathbf{x} = (x_1, x_2, \dots, x_n)$ and $\mathbf{y} = (y_1, y_2, \dots, y_n)$ be two vectors of $\mathbb{Z}^n$ with $x_1 < y_1$. Denote $\mathbf{y}' \triangleq (y_1 - 1, y_2, \dots, y_n)$. Then

$$|\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y})| \leqslant |\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y}')|.$$

Proof: Let $\mathbf{z} = (z_1, z_2, \dots, z_n)$ be an arbitrary vector of $(\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y})) \setminus (\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y}'))$. Since $x_1 < y_1$, we necessarily have $z_1 = y_1$ and the Hamming distance between $\mathbf{y}$ and $\mathbf{z}$ is exactly t. Let $\mathbf{z}' = (y_1 - 1, z_2, z_3, \dots, z_n)$. Then $\mathbf{z}' \in \mathcal{B}_t(\mathbf{y}')$ as $\mathbf{z} \in \mathcal{B}_t(\mathbf{y})$, but $\mathbf{z}' \notin \mathcal{B}_t(\mathbf{y})$ since $\mathbf{y}$ and $\mathbf{z}'$ are of Hamming distance $t+1$ apart. Furthermore, noting that $x_1 < y_1 = z_1$, we have $\mathbf{z}' \in \mathcal{B}_t(\mathbf{x})$. Hence, $\mathbf{z}' \in (\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y}')) \setminus (\mathcal{B}_t(\mathbf{x}) \cap \mathcal{B}_t(\mathbf{y}))$. Note that for different choices of $\mathbf{z}$, the resulting $\mathbf{z}'$ are pairwise distinct. Thus, we completed the proof. ■

Lemma 14: Assume that $\delta \leqslant t \leqslant n$ and $0 < k_- \leqslant k_+$. For any two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{Z}^n$ with $d_{k_+,k_-}(\mathbf{x}, \mathbf{y}) = \delta$, we have that

$$\sum_{i=0}^{t-\delta} \binom{n-2\delta}{i} (k_+ + k_-)^i \leqslant N(\mathbf{x}, \mathbf{y}; t, k_+, k_-) \leqslant \sum_{i=0}^{t-\delta} \binom{n}{i} (k_+ + k_-)^{i+2\delta}. \quad (9)$$

Proof: Let $n_1 = N_{k_-}(\mathbf{x}, \mathbf{y})$, $n_2 = N_{k_+,k_-}(\mathbf{x}, \mathbf{y})$, $m_1 = M_{k_+,k_-}(\mathbf{x}, \mathbf{y})$, and $m_2 = M_{k_+,k_-}(\mathbf{y}, \mathbf{x})$. According to Lemma 12 and Lemma 13, the maximal intersection is achieved if $\mathbf{x}$ and $\mathbf{y}$ have the following form:

$$\mathbf{x} = (\underbrace{0, 0, \dots, 0}_{n_1}, \underbrace{0, 0, \dots, 0}_{n_2}, \underbrace{k_- + 1, k_- + 1, \cdots, k_- + 1}_{m_1}, \underbrace{0, 0, \dots, 0}_{m_2}, \underbrace{0, 0, \dots, 0}_{n'})$$

and

$$\mathbf{y} = (\underbrace{1, 1, \dots, 1}_{n_1}, \underbrace{k_+ + 1, \dots, k_+ + 1}_{n_2}, \underbrace{0, 0, \dots, 0}_{m_1}, \underbrace{k_- + 1, \cdots, k_- + 1}_{m_2}, \underbrace{0, 0, \dots, 0}_{n'}).$$

where $n' = n - n_1 - n_2 - m_1 - m_2$. Partition the positions into the following five intervals:

$$\begin{aligned}
I_1 &= [1, n_1], \\
I_2 &= [n_1 + 1, n_1 + n_2], \\
I_3 &= [n_1 + n_2 + 1, n_1 + n_2 + m_1], \\
I_4 &= [n_1 + n_2 + m_1 + 1, n_1 + n_2 + m_1 + m_2], \\
I_5 &= [n - n' + 1, n].
\end{aligned}$$

Let $\mathbf{z}$ be an element in the intersection of the two balls $\mathcal{B}_t(\mathbf{x})$ and $\mathcal{B}_t(\mathbf{y})$. Obviously, $\mathbf{x}$ and $\mathbf{z}$ can have the same components only in the positions belonging to $I_1 \cup I_3 \cup I_5$, and $\mathbf{y}$ and $\mathbf{z}$ can have the same components only in the positions belonging to $I_1 \cup I_4 \cup I_5$. Denote by i the number of positions where $\mathbf{x}$ and $\mathbf{y}$ agree but differ from $\mathbf{z}$; so all these i positions come from I_5. Denote

$$\begin{aligned}
j &= |\{\ell \in I_1; \mathbf{z}[\ell] = \mathbf{x}[\ell]\}|, \\
k &= |\{\ell \in I_1; \mathbf{z}[\ell] = \mathbf{y}[\ell]\}|, \\
r &= |\{\ell \in I_3; \mathbf{z}[\ell] = \mathbf{x}[\ell]\}|, \\
s &= |\{\ell \in I_4; \mathbf{z}[\ell] = \mathbf{y}[\ell]\}|.
\end{aligned}$$

Then we have $j + k \leqslant n_1$, $r \leqslant m_1$, $s \leqslant m_2$, $n_1 - j + n_2 + m_1 - r + m_2 + i \leqslant t$, and $n_1 - k + n_2 + m_1 + m_2 - s + i \leqslant t$. Hence,

$$\begin{aligned}
i &\leqslant \min \left\{ t - \left\lceil \frac{n_1 + m_1 + m_2}{2} \right\rceil - n_2, t - m_1 - n_2, t - m_2 - n_2 \right\} \\
&= t - \max \left\{ \left\lceil \frac{n_1 + m_1 + m_2}{2} \right\rceil + n_2, m_1 + n_2, m_2 + n_2 \right\} \\
&= t - d_{k_+,k_-}(\mathbf{x}, \mathbf{y}).
\end{aligned}$$

Thus, the number of choices of $\mathbf{z}$ is at most

$$\begin{aligned}
&\sum_{i=0}^{t-\delta} \binom{n'}{i} (k_+ + k_-)^i \sum_j \binom{n_1}{j} \\
&\quad \times \sum_k \binom{n_1 - j}{k} (k_+ + k_- - 2)^{n_1 - j - k} (k_-)^{n_2} \\
&\quad \times \sum_r \binom{m_1}{r} \sum_s \binom{m_2}{s} (k_+)^{m_1 + m_2 - r - s} \\
&\leqslant \sum_{i=0}^{t-\delta} \binom{n'}{i} (k_+ + k_-)^{i + n_1} (k_-)^{n_2} (k_+ + 1)^{m_1 + m_2} \\
&< \sum_{i=0}^{t-\delta} \binom{n}{i} (k_+ + k_-)^{i + 2\delta}.
\end{aligned}$$

For the lower bound, we have that

$$\begin{aligned}
N(\mathbf{x}, \mathbf{y}; t, k_+, k_-) &\geqslant \sum_{i=0}^{t-\delta} \binom{n'}{i} (k_+ + k_-)^i \\
&\geqslant \sum_{i=0}^{t-\delta} \binom{n-2\delta}{i} (k_+ + k_-)^i.
\end{aligned}$$

■

Theorem 15: Let k_+, k_-, t and δ be fixed integers such that $0 < k_- \leqslant k_+$ and $1 \leqslant \delta \leqslant t$. For any code $\mathcal{C} \subseteq \mathbb{Z}^n$, $N(\mathcal{C}; t, k_+, k_-) = \Theta(n^{t-\delta})$ if and only if $\mathcal{C}$ can correct up to $\delta - 1$ (k_+, k_-)-limited-magnitude errors.

Proof: According to Lemma 14, $N(\mathcal{C}; t, k_+, k_-) = \Theta(n^{t-\delta})$ if and only if $d_{k_+,k_-}(\mathcal{C}) = \delta$. Combining this with Proposition 11, the theorem is proved. ■

IV. Single-Error Lattice Reconstruction Codes

In this section, we study the design of reconstruction codes from a given number received sequences. In other words, given a positive integer N, we would like to construct a code $\mathcal{C} \subseteq \mathbb{Z}^n$ such that $N(\mathcal{C}; t, k_+, k_-) \leqslant N$. Since the general problem seems to be involved, we focus on the case of lattice codes in the channel which introduces a single (k_+, k_-)-limited-magnitude error. Theorem 6 shows that if $N \geqslant k_+ + k_-$, we can take the whole space $\mathbb{Z}^n$ as our code, and its density is 1. In the other extremal case of $N = 0$, Theorem 4 and Definition 3 imply a sphere-packing bound on the density. That is, any lattice code correcting a single error should have density at most $(n(k_+ + k_-) + 1)^{-1}$, which is $O(1/n)$.

In the following, we study the case of $N = 1$, and since lattice codes are in question, we base our approach on group splitting.

Lemma 16: Assume that $1 \leqslant k_- \leqslant k_+$. Let $\Lambda \subseteq \mathbb{Z}^n$ be a lattice. Define $G = \mathbb{Z}^n/\Lambda$, and $s_i = \phi(\mathbf{e}_i)$ for $1 \leqslant i \leqslant n$, where $\phi : \mathbb{Z}^n \to G$ is the natural homomorphism. Let $\mathbf{s} = (s_1, s_2, \dots, s_n)$ and so $\Lambda = \{\mathbf{x} \in \mathbb{Z}^n; \mathbf{x} \cdot \mathbf{s} = 0\}$. Then $N(\Lambda; 1, k_+, k_-) \leqslant 1$ if and only if all the following hold:

(C1) $as_i \neq 0$ for all $1 \leqslant i \leqslant n$ and $a \in [-k_-, k_+]^*$.
(C2) $as_i \neq bs_i$ for all $1 \leqslant i \leqslant n$ and all distinct $a, b \in [-k_-, k_+]^*$, except $|a - b| = k_+ + k_-$.
(C3) $as_i \neq bs_j$ for all $1 \leqslant i < j \leqslant n$ and all $a, b \in [-k_-, k_-]^*$.

Proof: We first show that if $N(\Lambda; 1, k_+, k_-) \leqslant 1$, then the conditions hold.

1) For (C1), w.l.o.g., suppose to the contrary that $as_1 = 0$ for some $a \in [-k_-, k_+]^*$. Then both the vectors $\mathbf{x} = (0, 0, \dots, 0)$ and $\mathbf{y} = (a, 0, \dots, 0)$ belong to Λ. If $a \leqslant k_-$, then $\{\mathbf{x}, \mathbf{y}\} \subseteq \mathcal{B}_1(\mathbf{x}) \cap \mathcal{B}_1(\mathbf{y})$; otherwise, the intersection contains $(a, 0, \dots, 0)$ and $(a - 1, 0, \dots, 0)$.
2) For (C2), w.l.o.g., suppose to the contrary that $as_1 = bs_1$ for some $a, b \in [-k_-, k_-]^*$ with $b > a$ and $|a - b| < k_+ + k_-$. Consider the two codewords $\mathbf{x} = (0, 0, \dots, 0)$ and $\mathbf{y} = (b - a, 0, \dots, 0)$. Since $1 \leqslant b - a \leqslant k_+ + k_- - 1$, then $b - a - k_-, b - a - k_- + 1 \in [-k_-, k_+]$. Hence, the intersection $\mathcal{B}_1(\mathbf{x}) \cap \mathcal{B}_1(\mathbf{y})$ contains two vectors $(b - a - k_-, 0, \dots, 0)$ and $(b - a - k_- + 1, 0, \dots, 0)$, a contradiction.
3) For (C3), suppose to the contrary that $as_i = bs_j$ for some $1 \leqslant i < j \leqslant n$ and $a, b \in [-k_-, k_-]^*$. W.l.o.g., we assume that $i = 1$ and $j = 2$. Then both the vectors $\mathbf{x} = (0, 0, \dots, 0)$ and $\mathbf{y} = (a, -b, 0, \dots, 0)$ belong to Λ as $\mathbf{x} \cdot \mathbf{s} = \mathbf{y} \cdot \mathbf{s} = 0$. However, the intersection $\mathcal{B}_1(\mathbf{x}) \cap \mathcal{B}_1(\mathbf{y})$ contains two vectors $(a, 0, 0, \dots, 0)$ and $(0, -b, 0, \dots, 0)$, which contradicts $N(\Lambda; 1, k_+, k_-) \leqslant 1$.

Now, we show the other direction. Assume that $\mathbf{x}, \mathbf{y} \in \Lambda$, $\mathbf{x} \neq \mathbf{y}$, with $\mathcal{B}_1(\mathbf{x}) \cap \mathcal{B}_1(\mathbf{y}) \neq \varnothing$. Then $\mathbf{x} + a\mathbf{e}_i = \mathbf{y} + b\mathbf{e}_j$ for some $1 \leqslant i, j \leqslant n$ and $a, b \in [-k_-, k_+]$. Hence, $as_i - bs_j = (a\mathbf{e}_i - b\mathbf{e}_j) \cdot \mathbf{s} = (\mathbf{y} - \mathbf{x}) \cdot \mathbf{s} = 0$, and so $as_i = bs_j$. We consider the following two cases.

1) If $i \neq j$, according to (C1) and (C3), necessarily $a, b \neq 0$ and $\max\{a, b\} > k_-$. W.l.o.g., assume that $i = 1$, $j = 2$, and $b > k_-$. Then if $\mathbf{x} = (x_1, x_2, \dots, x_n)$, we have $\mathbf{y} = (x_1 + a, x_2 - b, x_3, \dots, x_n)$. Since $b > k_-$, the intersection of the two balls only contains a unique vector, i.e., $(x_1 + a, x_2, \dots, x_n)$.
2) If $i = j$, according to (C1) and (C2), necessarily $|b - a| = k_+ + k_-$. W.l.o.g., assume that $i = j = 1$ and $b - a = k_+ + k_-$. Then if $\mathbf{x} = (x_1, x_2, \dots, x_n)$, we have $\mathbf{y} = (x_1 - (k_+ + k_-), x_2, \dots, x_n)$. Thus the intersection only contains the vector $(x_1 - k_-, x_2, x_2, \dots, x_n)$.

■

Corollary 17: Assume that $1 \leqslant k_- \leqslant k_+$. Let $\Lambda \subseteq \mathbb{Z}^n$ be a lattice code such that $N(\Lambda; 1, k_+, k_-) \leqslant 1$. Then

$$|\mathbb{Z}^n/\Lambda| \geqslant \begin{cases} \max\{2nk_- + 1, k_+ + k_-\}, & \text{if } k_+ > k_-, \\ \max\{n(k_+ + k_- - 1) + 1, k_+ + k_-\}. & \text{if } k_+ = k_-. \end{cases}$$

Proof: Note that the conditions (C1)–(C3) are equivalent to the following two conditions:

(C1') $as_i \neq bs_i$ for all $1 \leqslant i \leqslant n$ and all distinct $a, b \in [-k_-, k_+ - 1]$.
(C2') $as_i \neq bs_j$ for all $1 \leqslant i < j \leqslant n$ and all $a, b \in [-k_-, k_-]$, except $a = b = 0$.

For any $\mathbf{x} \in \mathbb{Z}^n$, we can think of $\mathbf{x} \cdot \mathbf{s}$ as a syndrome. Thus, the elements of Λ are exactly those with the 0 syndrome, and the elements of a coset $\mathbf{v} + \Lambda$ are exactly those with syndrome $\mathbf{v} \cdot \mathbf{s}$. Hence,

$$|\mathbb{Z}^n/\Lambda| \geqslant |\{as_i; a \in [-k_-, k_+], 1 \leqslant i \leqslant n\}| .$$

The bound now follows by (C1') and (C2'). ■

From the bound above, we can see that if $k_- > 0$ and k_+, k_- are fixed, any lattice code with $N(\Lambda; 1, k_+, k_-) \leqslant 1$ has density at most $O(1/n)$, which is asymptotically the same as the case of $N(\Lambda; 1, k_+, k_-) = 0$.

For $k_- = 0$, however, it is much different. There are codes with $N(\Lambda; 1, k_+, 0) \leqslant 1$ having constant density. This is trivially true for $k_+ = 1$ since then the lattice $\Lambda = \mathbb{Z}^n$ has density 1 and it satisfies $N(\mathbb{Z}^n; 1, 1, 0) = 1$. For $k_+ \geqslant 2$ we have the following:

Lemma 18: Assume that $k_- = 0$ and $k_+ \geqslant 2$. Let $\Lambda \subseteq \mathbb{Z}^n$ be a lattice. Let $G \triangleq \mathbb{Z}^n/\Lambda$ and $s_i = \phi(\mathbf{e}_i)$ for $1 \leqslant i \leqslant n$, where $\phi : \mathbb{Z}^n \to G$ is the natural homomorphism. Let $\mathbf{s} = (s_1, s_2, \dots, s_n)$ and so $\Lambda = \{\mathbf{x} \in \mathbb{Z}^n; \mathbf{x} \cdot \mathbf{s} = 0\}$. Then $N(\Lambda; 1, k_+, 0) \leqslant 1$ if and only if $as_i \neq bs_i$ for all $1 \leqslant i \leqslant n$ and all distinct $a, b \in [0, k_+ - 1]$.

Proof: ($\Rightarrow$) Suppose to the contrary that $as_i = bs_i$ for some $a, b \in [0, k_+ - 1]$ with $b > a$. W.l.o.g., assume $i = 1$. Consider the two codewords $\mathbf{x} = (0, 0, \dots, 0)$ and $\mathbf{y} = (b - a, 0, \dots, 0)$. Since $1 \leqslant b - a \leqslant k_+ - 1$, the intersection $\mathcal{B}_1(\mathbf{x}) \cap \mathcal{B}_1(\mathbf{y})$ contains the two vectors $(b - a, 0, \dots, 0)$ and $(b - a + 1, 0, \dots, 0)$, a contradiction.

($\Leftarrow$) Assume that $\mathbf{x}, \mathbf{y} \in \Lambda$, $\mathbf{x} \neq \mathbf{y}$, with $\mathcal{B}_1(\mathbf{x}) \cap \mathcal{B}_1(\mathbf{y}) \neq \varnothing$. Then $\mathbf{x} + a\mathbf{e}_i = \mathbf{y} + b\mathbf{e}_j$ for some $1 \leqslant i, j \leqslant n$ and $a, b \in [0, k_+]$. So $as_i - bs_j = (a\mathbf{e}_i - b\mathbf{e}_j) \cdot \mathbf{s} = (\mathbf{y} - \mathbf{x}) \cdot \mathbf{s} = 0$, hence, $as_i = bs_j$. We consider the following two cases.

1) If $i \neq j$ and $a, b \neq 0$, w.l.o.g., assume that $i = 1$, $j = 2$ and $b \geqslant a$. Then if $\mathbf{x} = (x_1, x_2, \dots, x_n)$, we have $\mathbf{y} = (x_1 + a, x_2 - b, x_3, \dots, x_n)$. Since $a, b > 0$, the intersection of the two balls only contains the vector $(x_1 + a, x_2, \dots, x_n)$. If $a = 0$, then $\mathbf{x} + 0 \cdot \mathbf{e}_i = \mathbf{x} + 0 \cdot \mathbf{e}_j = \mathbf{x} = \mathbf{y} + b\mathbf{e}_j$, which is included in the case $i = j$, and a symmetric argument applies to the case of $b = 0$.

2) If $i = j$, according to our condition, $(a, b) = (0, k_+)$ or $(k_+, 0)$. In both cases, the intersection only contains one vector, i.e., $\mathbf{x}$ or $\mathbf{y}$ respectively. ■

Corollary 19: Assume that $k_- = 0$ and $k_+ \geqslant 2$. Let $\Lambda \subseteq \mathbb{Z}^n$ be a lattice code such that $N(\Lambda; 1, k_+, 0) \leqslant 1$. Then

$$|\mathbb{Z}^n/\Lambda| \geqslant k_+.$$

Moreover, the bound can be attained by letting $G = \mathbb{Z}_{k_+}$ and $\mathbf{s} = (1, 1, \ldots, 1)$, where the corresponding lattice code is

$$\Lambda = \left\{ (x_1, x_2, \ldots, x_n) \in \mathbb{Z}^n \middle| \sum_{i=1}^{n} x_i \equiv 0 \pmod{k_+} \right\}.$$

Proof: The bound comes directly from Lemma 18. For the code Λ, since $s_i = 1$, $as_i \not\equiv 0 \pmod{k_+}$ for all $a \in [1, k_+ - 1]$. Thus, according to Lemma 18, we have $N(\Lambda; 1, k_+, 0) \leqslant 1$. ■

Now, we study the case of $N(\Lambda; 1, k_+, k_-) \leqslant 2$ with $k_- \geqslant 1$. We have the following result which is similar to the case of $N(\Lambda; 1, k_+, 0) \leqslant 1$. This time, $(k_+, k_-) = (1, 1)$ is a trivial case in which we can take $\Lambda = \mathbb{Z}^n$ since $N(\mathbb{Z}^n; 1, 1, 1) = 2$. The non-trivial cases are given by the following:

Lemma 20: Assume that $1 \leqslant k_- \leqslant k_+$ and $k_+ + k_- \geqslant 3$. Let $\Lambda \subseteq \mathbb{Z}^n$ be a lattice. Let $G \triangleq \mathbb{Z}^n/\Lambda$ and $s_i = \phi(\mathbf{e}_i)$ for $1 \leqslant i \leqslant n$, where $\phi : \mathbb{Z}^n \to G$ is the natural homomorphism. Let $\mathbf{s} = (s_1, s_2, \ldots, s_n)$ and so $\Lambda = \{\mathbf{x} \in \mathbb{Z}^n; \mathbf{x} \cdot \mathbf{s} = 0\}$. Then $N(\Lambda; 1, k_+, k_-) \leqslant 2$ if and only if $as_i \neq bs_i$ for all $1 \leqslant i \leqslant n$ and all distinct $a, b \in [-k_-, k_+ - 2]$.

Proof: ($\Rightarrow$) Suppose to the contrary that $as_i = bs_i$ for some $a, b \in [-k_-, k_+ - 2]$. W.l.o.g., assume that $i = 1$ and $b > a$. Consider the two codewords $\mathbf{x} = (0, 0, \ldots, 0)$ and $\mathbf{y} = (b - a, 0, \ldots, 0)$. Since $1 \leqslant b - a \leqslant k_+ + k_- - 2$, the intersection $\mathcal{B}_1(\mathbf{x}) \cap \mathcal{B}_1(\mathbf{y})$ contains three vectors $(b - a - k_-, 0, \ldots, 0)$ and $(b - a - k_- + 1, 0, \ldots, 0)$ and $(b - a - k_- + 2, 0, \ldots, 0)$, a contradiction.

($\Leftarrow$) Assume that $\mathbf{x}, \mathbf{y} \in \Lambda$, $\mathbf{x} \neq \mathbf{y}$, with $\mathcal{B}_1(\mathbf{x}) \cap \mathcal{B}_1(\mathbf{y}) \neq \varnothing$. Then $\mathbf{x} + a\mathbf{e}_i = \mathbf{y} + b\mathbf{e}_j$ for some $1 \leqslant i, j \leqslant n$ and $a, b \in [-k_-, k_+]$. Hence, $as_i - bs_j = (a\mathbf{e}_i - b\mathbf{e}_j) \cdot \mathbf{s} = (\mathbf{y} - \mathbf{x}) \cdot \mathbf{s} = 0$, and so $as_i = bs_j$. We consider the following two cases:

1) If $i \neq j$ and $a, b \neq 0$, i.e., $\mathbf{x}$ and $\mathbf{y}$ differ in two positions, necessarily the intersection of the two balls contains two vectors if $a, b \in [-k_-, k_-]$, or contains one vector otherwise. If either $a = 0$ or $b = 0$ then the case is covered by the following case of $i = j$.
2) If $i = j$, according to our assumption, $|a - b| = k_+ + k_- - 1$ or $k_+ + k_-$. If $|a - b| = k_+ + k_-$, the intersection only contains one vector, and if $|a - b| = k_+ + k_- - 1$, the intersection contains two vectors. ■

Corollary 21: Assume that $1 \leqslant k_- \leqslant k_+$ and $k_+ + k_- \geqslant 3$. Let $\Lambda \subseteq \mathbb{Z}^n$ be a lattice code such that $N(\Lambda; 1, k_+, k_-) \leqslant 2$. Then

$$|\mathbb{Z}^n/\Lambda| \geqslant k_+ + k_- - 1.$$

Moreover, the bound can be attained by letting $G = \mathbb{Z}_{k_+ + k_- - 1}$ and $\mathbf{s} = (1, 1, \ldots, 1)$, where the corresponding code is

$$\Lambda = \left\{ (x_1, \ldots, x_n) \in \mathbb{Z}^n \middle| \sum_{i=1}^{n} x_i \equiv 0 \pmod{k_+ + k_- - 1} \right\}.$$

V. Efficient Reconstruction Algorithms

In this section, we present two reconstruction algorithms for the (k_+, k_-)-limited-magnitude errors. We assume nothing about the structure of the code. In particular, we do not assume the codes are linear, i.e., lattice codes. Since more errors may occur in the received vectors than that are correctable by unique decoding, our strategy is to combine the received vectors into a single vector that is guaranteed to be within the unique-decoding radius from the transmitted codeword, and then use a unique decoding procedure. We thus reduce the reconstruction problem to a classical decoding problem.

If $\mathcal{C} \subseteq \mathbb{Z}^n$ is a code capable of correcting $\delta - 1$ (k_+, k_-)-limited-magnitude errors, we assume the existence of a decoding function $\mathcal{D}_{\mathcal{C}} : \mathbb{Z}^n \to \mathcal{C}$ which upon receiving a codeword corrupted by at most $\delta - 1$ (k_+, k_-)-limited-magnitude errors, is capable of finding the transmitted codeword. If there exists an efficient test of whether a vector is in $\mathcal{C}$, as in the case of lattice codes, then a naive brute-force implementation of a decoding procedure is possible in time complexity $O(|\mathcal{B}(n, \delta - 1, k_+, k_-)|) = O(n^{\delta - 1})$ by testing all the vectors in a ball centered at the received vector.

The first algorithm we present only works for the case of $k_- = 0$ and requires a few more received vectors than the upper bound of Lemma 8. However, it is quite simple.

Theorem 22: Let $\mathcal{C} \subseteq \mathbb{Z}^n$ be a code with the minimum distance $d_{k_+}(\mathcal{C}) = \delta$. Denote

$$\begin{aligned} N_a &\triangleq (k_+)^{\delta} \sum_{i=0}^{t-\delta} \binom{n-\delta}{i} (k_+)^i + 1 \\ &= (k_+)^{\delta} V_{k_+ + 1}(n - \delta, t - \delta) + 1. \end{aligned}$$

Let $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{N_a}$ be N_a distinct vectors that come from the same ball $\mathbf{x} + \mathcal{B}(n, t, k_+, 0)$ for some codeword $\mathbf{x} \in \mathcal{C}$. Then we can reconstruct $\mathbf{x}$ from $\mathcal{Y} \triangleq \{\mathbf{y}_1, \ldots, \mathbf{y}_{N_a}\}$ with time complexity $O(nN_a + C)$, where C is the time complexity of the unique-decoding algorithm of $\mathcal{C}$.

Proof: Our reconstruction algorithm is summarized in Algorithm 1. Since $z_i = \min\{\mathbf{y}_1[i], \mathbf{y}_2[i], \ldots, \mathbf{y}_{N_a}[i]\}$ for each $i \in [1, n]$, it is easy to see that $0 \leqslant z_i - \mathbf{x}[i] \leqslant k_+$. Thus, $\mathbf{z} \in \mathbf{x} + \mathcal{B}(n, r, k_+, 0)$ for some integer r. Assume to the contrary that there are δ positions where all the vectors in $\mathcal{Y}$ differ from $\mathbf{x}$ on each of these positions. However, the number of such vectors is at most $(k_+)^{\delta} \sum_{i=0}^{t-\delta} \binom{n-\delta}{i} (k_+)^i$, which is strictly less than N_a. Thus, we can find $r \leqslant \delta - 1$, and $\mathbf{z} \in \mathbf{x} + \mathcal{B}(n, \delta - 1, k_+, 0)$. Since $\mathcal{C}$ has distance δ, we can run the unique-decoding algorithm, $\mathcal{D}_{\mathcal{C}}$, on $\mathbf{z}$ to recover $\mathbf{x}$. ■

Remark: The number of required vectors in Algorithm 1 is larger than the upper bound of Lemma 8 by, at most, a constant multiplicative factor, since the inner expression of $\sum_{k=\delta+i-1}^{t-i} \binom{\delta}{k} (k_+ - 1)^{\delta - k}$ in the upper bound of Lemma 8 is replaced here with k_+^{δ}.

Algorithm 1 Reconstruction Algorithm for $k_- = 0$

Input: an N_a-set $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{N_a}\} \subseteq \mathbf{x} + \mathcal{B}(n, t, k_+, 0)$ for some $\mathbf{x} \in \mathcal{C}$
Output: the codeword $\mathbf{x} \in \mathcal{C}$
for $1 \leqslant i \leqslant n$ **do**
    $z_i \leftarrow \min\{\mathbf{y}_1[i], \mathbf{y}_2[i], \dots, \mathbf{y}_{N_a}[i]\}$
end for
$\mathbf{z} \leftarrow (z_1, z_2, \dots, z_n)$
$\mathbf{x} \leftarrow \mathcal{D}_{\mathcal{C}}(\mathbf{z})$
return $\mathbf{x}$

Algorithm 2 Majority Algorithm

Input: an N_b-set $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{N_b}\} \subseteq \mathbb{Z}^n$ and a threshold τ
Output: a word $\mathbf{z} \in (\mathbb{Z} \cup \{?\})^n$
for $1 \leqslant i \leqslant n$ **do**
    $\mathcal{Y}_i \leftarrow \{\mathbf{y}_1[i], \mathbf{y}_2[i], \dots, \mathbf{y}_{N_b}[i]\}$ (multiset)
    $M_i \leftarrow \mathrm{Maj}(\mathcal{Y}_i)$
    if $2n_{M_i}(\mathcal{Y}_i) - N > \tau$ **then**
        $\mathbf{z}[i] \leftarrow M_i$
    else
        $\mathbf{z}[i] \leftarrow ?$
    end if
end for
return $\mathbf{z}$

In the following, we consider the case of $k_- > 0$. Our method is to modify the reconstruction algorithm from [1] which was suggested for a channel with substitutions.

For a finite multiset $\mathcal{M}$ with elements from $\mathbb{Z}$, denote $n_i(\mathcal{M})$ the number of times that the element i appears in $\mathcal{M}$. Denote $\mathrm{Maj}(\mathcal{M})$ the element which appears most frequently in $\mathcal{M}$. If there is more than one such element, we take the smallest one.

Let $\mathcal{C} \subseteq \mathbb{Z}^n$ be a code with $d_{k_+,k_-}(\mathcal{C}) = \delta$. Let $\mathbf{x} \in \mathcal{C}$ be a codeword and $\mathcal{Y}$ be a subset of $\mathbf{x} + \mathcal{B}(n, t, k_+, k_-)$ with N_b vectors, where

$$\begin{aligned} N_b &\triangleq \sum_{i=0}^{t-\delta} \binom{n}{i} (k_+ + k_-)^{i+2\delta} + 1 \\ &= (k_+ + k_-)^{2\delta} V_{k_+ + k_- + 1}(n, t-\delta) + 1. \end{aligned} \tag{10}$$

Denote

$$\begin{aligned} \tau &\triangleq \left(1 - \frac{2}{\delta}\right) N_b + \frac{2}{\delta} \sum_{i=0}^{t-\delta} \binom{n-\delta}{i} (k_+ + k_-)^{i+\delta} \\ &= \left(1 - \frac{2}{\delta}\right) N_b + \frac{2(k_+ + k_-)^{\delta}}{\delta} V_{k_+ + k_- + 1}(n-\delta, t-\delta). \end{aligned} \tag{11}$$

It is easy to check that $\tau < N_b$.

We apply the following majority algorithm (Algorithm 2) with threshold τ on $\mathcal{Y}$ to get an estimate $\mathbf{z}$ of $\mathbf{x}$. The returned estimate may also contain the symbol ? which indicates an erasure.

We have the following upper bounds on the number of errors and erasures in the estimate $\mathbf{z}$.

Lemma 23: Let $\mathcal{Y} \subseteq \mathbf{x} + \mathcal{B}(n, t, k_+, k_-)$ be an N_b-set, $\mathbf{x} \in \mathbb{Z}^n$, and let $\mathbf{z}$ be the output of Algorithm 2 when run on $\mathcal{Y}$ with τ from (11). Then $\mathbf{z}$ contains at most $\delta - 1$ errors compared with $\mathbf{x}$, that is,

$$|\{i \in [1, n]; \mathbf{x}[i] \neq \mathbf{z}[i], \mathbf{z}[i] \in \mathbb{Z}\}| \leqslant \delta - 1.$$

Proof: Suppose to the contrary that there are at least δ errors. W.l.o.g., we assume that the first δ symbols of $\mathbf{z}$ are erroneous. Let $M \triangleq [-k_-, k_+]^*$. For each $i \in [1, n]$ and $k \in M$, let

$$e_i^k = |\{\ell \in [1, N_b]; \mathbf{y}_\ell[i] = \mathbf{x}[i] + k\}|.$$

Then there is an error in the i-th position of $\mathbf{z}$ only if there is a $k \in M$ such that $2e_i^k - N_b > \tau$. It follows that

$$\begin{aligned} \sum_{i=1}^{\delta} \sum_{k \in M} e_i^k &> \delta \frac{N_b + \tau}{2} \\ &= (\delta - 1) N_b + \sum_{i=0}^{t-\delta} \binom{n-\delta}{i} (k_+ + k_-)^{i+\delta}. \end{aligned} \tag{12}$$

On the other hand, there are at most $\sum_{i=0}^{t-\delta} \binom{n-\delta}{i} (k_+ + k_-)^{i+\delta}$ vectors in $\mathcal{Y}$ that can have erroneous components in all of the first δ positions. For the other vectors in $\mathcal{Y}$, each has at most $\delta - 1$ errors in the first δ positions. Therefore,

$$\begin{aligned} \sum_{i=1}^{\delta} \sum_{k \in M} e_i^k &\leqslant \delta \sum_{i=0}^{t-\delta} \binom{n-\delta}{i} (k_+ + k_-)^{i+\delta} \\ &\quad + (\delta - 1) \left(N_b - \sum_{i=0}^{t-\delta} \binom{n-\delta}{i} (k_+ + k_-)^{i+\delta} \right) \\ &= (\delta - 1) N_b + \sum_{i=0}^{t-\delta} \binom{n-\delta}{i} (k_+ + k_-)^{i+\delta}, \end{aligned}$$

which contradicts (12). ■

Lemma 24: Let $\mathcal{Y} \subseteq \mathbf{x} + \mathcal{B}(n, t, k_+, k_-)$ be an N_b-set, $\mathbf{x} \in \mathbb{Z}^n$, and let $\mathbf{z}$ be the output of Algorithm 2 when run on $\mathcal{Y}$ with τ from (11). Then $\mathbf{z}$ contains at most $2t\delta$ erasures.

Proof: Let e_i^k be as in the proof of Lemma 23. Noting that $\mathbf{z}[i] = \mathbf{x}[i]$ if and only if $N_b - 2\sum_{k \in M} e_i^k > \tau$, there is an erasure on the i-th positions only if $\sum_{k \in M} e_i^k \geqslant \frac{N_b - \tau}{2}$. Since each vector $\mathbf{y}_\ell$ has at most t errors, the number of erasures is at most

$$\begin{aligned} \frac{2tN_b}{N_b - \tau} &= \frac{2tN_b}{\frac{2}{\delta}\left(N_b - \sum_{i=0}^{t-\delta} \binom{n-\delta}{i} (k_+ + k_-)^{i+\delta}\right)} \\ &\leqslant \frac{2tN_b}{\frac{2}{\delta}\left(N_b - N_b/(k_+ + k_-)^{\delta}\right)} \leqslant 2t\delta. \end{aligned}$$

■

Now we can present our reconstruction process in Algorithm 3.

Theorem 25: Let $\mathcal{C} \subseteq \mathbb{Z}^n$ be a code with the minimum distance $d_{k_+,k_-}(\mathcal{C}) = \delta$. Let N_b and τ be defined as in (10) and (11). Let $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{N_b}\}$ be a set of N_b

Algorithm 3 Reconstruction algorithm for $k_- > 0$

Input: an N_b-set $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{N_b}\} \subseteq \mathbf{x} + \mathcal{B}(n,t,k_+,k_-)$ for some $\mathbf{x} \in \mathcal{C}$, and a threshold τ

Output: the codeword $\mathbf{x} \in \mathcal{C}$

1: $\mathbf{z} \leftarrow$ the output of Algorithm 2 with $\mathcal{Y}$ and τ being the input
2: $\mathcal{E} \leftarrow \{i \in [1,n]; \mathbf{z}[i] = ?\}$
3: $\mathcal{U} \leftarrow \{\mathbf{u} \in \mathbb{Z}^n; \mathbf{u}[i] = \mathbf{z}[i]$ for all $i \notin \mathcal{E}$ and $\mathbf{u}[i] \in [\mathbf{y}_1[i] - k_+, \mathbf{y}_1[i] + k_-]$ for all $i \in \mathcal{E}\}$
4: **for** $\mathbf{u} \in \mathcal{U}$ **do**
5: $\quad \mathbf{x} \leftarrow \mathcal{D}_{\mathcal{C}}(\mathbf{u})$
6: $\quad$ **if** $\mathcal{Y} \subseteq \mathbf{x} + \mathcal{B}(n,t,k_+,k_-)$ **then**
7: $\quad\quad$ **return** $\mathbf{x}$
8: $\quad$ **end if**
9: **end for**

vectors coming from the same ball $\mathbf{x} + \mathcal{B}(n,t,k_+,0)$ for some codeword $\mathbf{x} \in \mathcal{C}$. Then we can reconstruct $\mathbf{x}$ by applying Algorithm 3 on $\mathcal{Y}$ and τ with time complexity $O(nN_b + C)$, where C is the time complexity of the unique-decoding algorithm of $\mathcal{C}$.

Proof: Let $\mathcal{U}$ be defined as in Algorithm 3. According to Lemma 23, there is a vector $\mathbf{u} \in \mathcal{U}$ such that $\mathbf{u} \in \mathbf{x} + \mathcal{B}(n, \delta-1, k_+, k_-)$. Thus we could apply the decoder $\mathcal{D}_{\mathcal{C}}$ of $\mathcal{C}$ to each vector of $\mathcal{U}$ to obtain a subset $\mathcal{S} \subseteq \mathcal{C}$ which contains $\mathbf{x}$. Finally, since $|\mathcal{Y}| = N_b > N(\mathcal{C}; t, k_+, k_-)$, the vector $\mathbf{x}$ can be identified from $\mathcal{S}$ by checking each codeword $\mathbf{c} \in \mathcal{S}$ whether $\mathcal{Y} \subseteq \mathbf{c} + \mathcal{B}(n,t,k_+,k_-)$.

Let us analyze the time complexity of Algorithm 3. The complexity of Step 1 is $O(nN_b)$. According to Lemma 24, the decoding loop starting in Step 4 takes at most $(k_+ + k_- + 1)^{2t\delta}$ rounds, which is independent of n. The complexity of checking the condition in Step 6 is also $O(nN_b)$. So the total time complexity of this algorithm is $O(nN_b + C)$. ∎

VI. List Decoding With Multiple Received Sequences

For a code $\mathcal{C} \subseteq \mathbb{Z}^n$ with minimum distance $d_{k_+,k_-}(\mathcal{C}) = \delta$, denote $f \triangleq t - \delta + 1$, that is, the number of excessive errors that $\mathcal{C}$ cannot cope with. Let $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$ be a subset of $\mathbf{x} + \mathcal{B}(n,t,k_+,k_-)$ for some $\mathbf{x} \in \mathcal{C}$. In Section III and Section V, we have shown that $\mathbf{x}$ can be recovered from $\mathcal{Y}$ if

$$N > \begin{cases} (k_+)^{\delta} V_{k_++1}(n-\delta, f-1), & \text{if } k_- = 0, \\ (k_+ + k_-)^{2\delta} V_{k_++k_-+1}(n, f-1), & \text{otherwise.} \end{cases} \tag{13}$$

In this section, we introduce another degree of freedom into our setting, which is the decoder's ability to return a list of codewords instead of a single one. We show that by doing so, the decoder requires substantially fewer vectors from the channel, compared with (13). We shall modify the reconstruction algorithms in Section V to produce a list of candidates $\mathcal{L}$ which contains the transmitted codeword $\mathbf{x}$. We first look at the case of $k_- = 0$.

Theorem 26: Let $\mathcal{C} \subseteq \mathbb{Z}^n$ be a code with the minimum distance $d_{k_+}(\mathcal{C}) = \delta$. Let $\mathbf{x} \in \mathcal{C}$ be a codeword and $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$ be an N-subset of $\mathbf{x} + \mathcal{B}(n,t,k_+,0)$. If

$$N > (k_+)^{\delta+a} V_{k_++1}(n-\delta-a, f-1-a),$$

where $0 \leqslant a \leqslant f-1$, then we can decode to get a list $\mathcal{L} \subseteq \mathcal{C}$ containing $\mathbf{x}$ with size

$$|\mathcal{L}| \leqslant V_{k_++1}(n,a).$$

Moreover, the time complexity of the decoding is $O(nN + n^a C)$, where C is the time complexity of the decoding algorithm of $\mathcal{C}$.

Proof: Let $\mathbf{z} \in \mathbb{Z}^n$ be defined by $\mathbf{z}[i] = \min\{\mathbf{y}_1[i], \mathbf{y}_2[i], \dots, \mathbf{y}_N[i]\}$ for each $i \in [1,n]$. Since $N > (k_+)^{\delta+a} V_{k_++1}(n-\delta-a, f-1-a)$, similarly to the proof of Theorem 22, we can show that $\mathbf{z} \in \mathbf{x} + \mathcal{B}(n, \delta-1+a, k_+, 0)$. Then there is a vector $\mathbf{u} \in \mathbf{z} - \mathcal{B}(n,a,k_+,0)$ such that $\mathbf{u} \in \mathbf{x} + \mathcal{B}(n,\delta-1,k_+,0)$. Thus we may apply the decoding of $\mathcal{C}$ on each vector of $\mathbf{z} - \mathcal{B}(n,a,k_+,0)$ to get the list $\mathcal{L}$, i.e.,

$$\mathcal{L} \triangleq \{\mathcal{D}_{\mathcal{C}}(\mathbf{u}); \mathbf{u} \in \mathbf{z} - \mathcal{B}(n,a,k_+,0)\}.$$

It also follows that $|\mathcal{L}| \leqslant V_{k_++1}(n,a)$. The claimed complexity follows from the fact that $V_{k_++1}(n,a) = O(n^a)$. ∎

Now we study the case of $k_- > 0$. Let $0 \leqslant a \leqslant f-1 = t-\delta$. Assume that

$$N \triangleq (k_+ + k_-)^{\delta+a+1} V_{k_++k_-+1}(n-\delta-a, f-1-a) + 1. \tag{14}$$

Denote

$$\tau \triangleq \left(1 - \frac{2}{\delta+a}\right) N + \frac{2}{\delta+a} \sum_{i=0}^{t-\delta-a} \binom{n-\delta-a}{i} (k_+ + k_-)^{i+\delta+a}. \tag{15}$$

Given an N-set $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\} \subseteq \mathbf{x} + \mathcal{B}(n,t,k_+,k_-)$ for some vector $\mathbf{x}$, we first apply Algorithm 2 with threshold τ on it to obtain an estimate $\mathbf{z}$ of $\mathbf{x}$. Similar to Lemma 23 and Lemma 24, we have the following result on $\mathbf{z}$. The proofs are the same as those in Section V and we omit here.

Lemma 27: Let $\mathcal{Y} \subseteq \mathbf{x} + \mathcal{B}(n,t,k_+,k_-)$ be an N-set, $\mathbf{x} \in \mathbb{Z}^n$, and let $\mathbf{z}$ be the output of Algorithm 2 when run on $\mathcal{Y}$ with τ from (15). Then $\mathbf{z}$ contains at most $\delta+a-1$ errors compared with $\mathbf{x}$, and at most $2t(\delta+a)$ erasures.

Proof: The proof is the same as those of Lemma 23 and Lemma 24. ∎

Our list decoding algorithm is presented in Algorithm 4.

Theorem 28: Let $\mathcal{C} \subseteq \mathbb{Z}^n$ be a code with the minimum distance $d_{k_+,k_-}(\mathcal{C}) = \delta$. Let N and τ be defined as in (14) and (15). Let $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$ be a set of N vectors contained in the same ball $\mathbf{x} + \mathcal{B}(n,t,k_+,0)$ for some codeword $\mathbf{x} \in \mathcal{C}$. Then we can decode a list $\mathcal{L}$ containing $\mathbf{x}$ by applying Algorithm 4 on $\mathcal{Y}$ and τ, where

$$|\mathcal{L}| \leqslant (k_+ + k_- + 1)^{2t(\delta+a)} V_{k_++k_-+1}(n,a).$$

The time complexity is $O(Nn + n^a C)$, where C is the time complexity of the decoding algorithm of $\mathcal{C}$.

Algorithm 4 List decoding algorithm

Input: an N-set $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N\} \subseteq \mathbf{x} + \mathcal{B}(n, t, k_+, k_-)$ for some $\mathbf{x} \in \mathcal{C}$, and a threshold τ

Output: a set $\mathcal{L} \subseteq \mathcal{C}$ such that $\mathbf{x} \in \mathcal{L}$

1: $\mathbf{z} \leftarrow$ the output of Algorithm 2 with $\mathcal{Y}$ and τ being the input

2: $\mathcal{E} \leftarrow \{i \in [1, n]; \mathbf{z}[i] = ?\}$

3: $\mathcal{U} \leftarrow \{\mathbf{u} \in \mathbb{Z}^n; \mathbf{u}[i] = \mathbf{z}[i]$ for all $i \notin \mathcal{E}$ and $\mathbf{u}[i] \in [\mathbf{y}_1[i] - k_+, \mathbf{y}_1[i] + k_-]$ for all $i \in \mathcal{E}\}$

4: $\mathcal{V} \leftarrow \bigcup_{\mathbf{u} \in \mathcal{U}} (\mathbf{u} - \mathcal{B}(n, a, k_+, k_-))$

5: $\mathcal{L} \leftarrow \{\mathcal{D}_{\mathcal{C}}(\mathbf{v}); \mathbf{v} \in \mathcal{V}\}$

6: **return** $\mathcal{L}$

Proof: Let $\mathcal{U}$ be defined as in Algorithm 4. According to Lemma 27, there is a vector $\mathbf{u} \in \mathcal{U}$ such that $\mathbf{u} \in \mathbf{x} + \mathcal{B}(n, \delta + a - 1, k_+, k_-)$. Since $\mathcal{V} = \bigcup_{\mathbf{u} \in \mathcal{U}} (\mathbf{u} - \mathcal{B}(n, a, k_+, k_-))$, we can find a vector $\mathbf{v} \in \mathcal{V}$ such that $\mathbf{v} \in \mathbf{x} + \mathcal{B}(n, \delta - 1, k_+, k_-)$. Thus we can apply the decoder $\mathcal{D}_{\mathcal{C}}$ of $\mathcal{C}$ to each vector of $\mathcal{V}$ to obtain a subset $\mathcal{L} \subseteq \mathcal{C}$ which contains $\mathbf{x}$. Moreover, the size of list

$$
\begin{aligned}
|\mathcal{L}| &\leqslant |\mathcal{V}| \leqslant |\mathcal{U}| V_{k_+ + k_- + 1}(n, a) \\
&\leqslant (k_+ + k_- + 1)^{2t(\delta + a)} V_{k_+ + k_- + 1}(n, a),
\end{aligned}
$$

where the last inequality holds as Lemma 27 implies that $|\mathcal{U}| \leqslant (k_+ + k_- + 1)^{2t(\delta + a)}$.

Let us analyze the time complexity of Algorithm 4. The complexity of Step 1 is $O(nN)$. The upper bound on $|\mathcal{L}|$ also bounds the number of times we run $\mathcal{D}_{\mathcal{C}}$, hence, we use the unique-decoder for $\mathcal{C}$ at most $O(n^a)$ times. Thus, the total time complexity of this algorithm is $O(nN + n^a C)$. ■

In [10], the trade-off between the size of the minimum list and the number of different received noisy sequences has been analyzed for substitutions. Modifying the approach therein, we could give another list-decoding algorithm which reduces simultaneously the value of N and the size of $\mathcal{L}$ in Theorem 28, at the cost of the time complexity. We require the following q-ary Sauer-Shelah lemma.

Lemma 29 ([6]): For all integers q, n, c with $c \leqslant n$, for any set $\mathcal{S} \subseteq [0, q-1]^n$, if $|\mathcal{S}| > V_q(n, c-1)$, then there exists some set of coordinates $U \subseteq [1, n]$ with $|U| = c$ such that for every $\mathbf{u} \in [0, q-1]^U$, there exists some $\mathbf{v} \in \mathcal{S}$ such that $\mathbf{u}$ and $\mathbf{v}|_U$ differ in every position.

Remark: The original condition in [6] is $|\mathcal{S}| > 2((q-1)n)^{c-1}$. However, even if we replace it with $|\mathcal{S}| > V_q(n, c-1)$, the proof still works and so the conclusion still holds.

Theorem 30: Let $\mathcal{C} \subseteq \mathbb{Z}^n$ be a code with $d_{k_+, k_-}(\mathcal{C}) = \delta$. Let $\mathbf{x} \in \mathcal{C}$ be a codeword and $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N\}$ be a subset of $\mathbf{x} + \mathcal{B}(n, t, k_+, k_-)$ of size N. If $N > V_{k_+ + k_- + 1}(n, f - 1 - a)$ where $0 \leqslant a \leqslant f - 1$, then we can decode to get a list $\mathcal{L}$ containing $\mathbf{x}$ with size

$$
|\mathcal{L}| \leqslant (k_+ + k_- + 1)^{2(f-a)} V_{k_+ + k_- + 1}(n - f + a, a).
$$

Proof: For each $i \in [1, n]$, define

$$
\begin{aligned}
m_i &\triangleq \min\{\mathbf{y}[i]; \mathbf{y} \in \mathcal{Y}\}, \\
M_i &\triangleq \max\{\mathbf{y}[i]; \mathbf{y} \in \mathcal{Y}\}, \\
K_i &\triangleq [\min\{m_i, M_i - k_+\}, \max\{M_i, m_i + k_-\}].
\end{aligned}
$$

Then $|K_i| \leqslant k_+ + k_- + 1$ for all $i \in [1, n]$. Furthermore, we have

$$
\mathbf{x} \in K_1 \times K_2 \times \cdots \times K_n
$$

and

$$
\mathbf{y}_\ell \in K_1 \times K_2 \times \cdots \times K_n
$$

for every $\ell \in [1, N]$. Since $N > V_{k_+ + k_- + 1}(n, f - 1 - a)$, according to Lemma 29, there is a subset $U \subseteq [1, n]$ of size $f - a$ and a vector $\mathbf{y}_{\ell_0} \in \mathcal{Y}$ such that $\mathbf{x}$ differs from $\mathbf{y}_{\ell_0}$ in every position of U.

Let $\mathcal{Y}'$ be a minimal subset of $\mathcal{Y}$ such that for every $\mathbf{y} \in \mathcal{Y}$ there is a vector $\mathbf{y}' \in \mathcal{Y}'$ with $\mathbf{y}$ and $\mathbf{y}'$ being the same on U. Then $|\mathcal{Y}'| \leqslant (k_+ + k_- + 1)^{|U|}$, and there is also a vector $\mathbf{y}^* \in \mathcal{Y}'$ such that $\mathbf{x}$ differs from $\mathbf{y}^*$ in every position of U. Let

$$
\mathcal{D} \triangleq \bigcup_{\mathbf{y}' \in \mathcal{Y}'} \{\mathbf{z} \in \mathbf{y}' - \mathcal{B}(n, f, k_+, k_-) \mid \mathbf{z}[i] \neq \mathbf{y}'[i] \text{ for every } i \in U\}.
$$

Since $\mathbf{y}' \in \mathbf{x} + \mathcal{B}(n, t, k_+, k_-)$ for all $\mathbf{y}' \in \mathcal{Y}'$ and $f = t - \delta + 1$, there is a vector $\mathbf{z}^* \in \mathcal{D}$ such that $\mathbf{z}^*$ and $\mathbf{x}$ agree in every position of U and $\mathbf{z}^* \in \mathbf{x} + \mathcal{B}(n, \delta - 1, k_+, k_-)$. Let

$$
\mathcal{L} = \{\mathcal{D}_{\mathcal{C}}(\mathbf{z}); \mathbf{z} \in \mathcal{D}\},
$$

where $\mathcal{D}_{\mathcal{C}}$ is the decoder of $\mathcal{C}$. Then $\mathbf{x} \in \mathcal{L}$ and the size

$$
\begin{aligned}
|\mathcal{L}| &\leqslant |\mathcal{D}| \leqslant |\mathcal{Y}'| (k_+ + k_-)^{|U|} V_{k_+ + k_- + 1}(n - |U|, f - |U|) \\
&< (k_+ + k_- + 1)^{2(f-a)} V_{k_+ + k_- + 1}(n - f + a, a).
\end{aligned}
$$

■

To the best of our knowledge, there is no efficient algorithm to identify the subset $\mathcal{U}$ in the Sauer-Shelah lemma. A brute-force algorithm requires $O(n^{f-a} N)$ comparisons. Thus the total time complexity of the decoding is $O(n^{f-a} N + n^a C)$, whereas the complexity of Algorithm 4 is $O(nN + n^a C)$.

Theorem 30 requires at least $V_{k_+ + k_- + 1}(n, f - 1 - a) + 1$ distinct received sequences to obtain a list of size $O(n^a)$. A natural question that arises is whether this requirement is tight? The following lemma, modified from [10, Lemma 32], shows that it is almost tight. The lemma shows that if $N \leqslant V_{k_+ + k_-}(n, f - 1 - a)$, there is a code $\mathcal{C} \subseteq \mathbb{Z}^n$, and a list $\mathcal{L} \subseteq \mathcal{C}$ of size $\Omega(n^{a+1})$ such that $\mathcal{Y} \subseteq \bigcap_{\mathbf{u} \in \mathcal{L}} (\mathbf{u} + \mathcal{B}(n, t, k_+, k_-))$, i.e., $\mathcal{Y}$ is in the intersection of too many balls around codewords.

Lemma 31: Assume that $(k_+, k_-) \neq (1, 0)$. Let $N \leqslant V_{k_+ + k_-}(n, f - a)$, where $0 \leqslant a \leqslant f$ and $n \geqslant 2e + a + 1$. Then there is a set of vectors $\mathcal{Y} \subseteq \mathbb{Z}^n$ with $|\mathcal{Y}| = N$ and a code $\mathcal{C} \subseteq \mathbb{Z}^n$ of size

$$
|\mathcal{C}| \geqslant \frac{n^a}{(e + a)^a \sum_{i=0}^{e} \binom{e+a}{i}},
$$

such that

- $\mathcal{C}$ can correct e (k_+, k_-)-limited-magnitude errors; and
- $\mathcal{Y} \subseteq \bigcap_{\mathbf{x}\in\mathcal{C}} (\mathbf{x} + \mathcal{B}(n,t,k_+,k_-))$.

Proof: Let $\mathcal{S} = \{\mathbf{v} \in [-k_-, k_+-1]^n; \mathrm{wt}(\mathbf{v}) \leqslant f-a\}$ and $\mathcal{Y}$ be an arbitrary subset of $\mathcal{S}$ with $|\mathcal{Y}| = N$. Let $\mathcal{C} \subseteq \{-1, 0\}^n$ be a binary code with minimum Hamming distance $2e+2$ and constant weight $e+a$. The lower bound on the size of $\mathcal{C}$ comes from the Gilbert-Varshamov bound, details of which can be found in the proof of [10, Lemma 32]. Since $\mathcal{C}$ can correct e substitutions, it also can correct the same number of (k_+, k_-)-limited-magnitude errors. Noting that $e+a+f-a = t$, we have that $\mathcal{Y} \subseteq \bigcap_{\mathbf{x}\in\mathcal{C}} (\mathbf{x} + \mathcal{B}(n,t,k_+,k_-))$. ■

VII. Reconstruction for Uniform Tandem Duplications

In this section, we show that our reconstruction algorithm for $(k_+, 0)$-limited-magnitude errors (Algorithm 1) can also be used for *tandem duplications*, which create a copy of a block of the sequence and insert it in a tandem manner, i.e., next to the original. For example, after a tandem duplication of length 3, the sequence 01032 may become 0103$\underline{103}$2, where the copy is underlined.

The design of reconstruction codes against t tandem duplications of the same length k was studied in [31]. Such a code could be decomposed into a family of subcodes $\mathcal{C}_{\mathbf{x}}$'s, so that the codewords from the same subcode shares the same *root* $\mathbf{x}$. This vector $\mathbf{x}$ could be computed from the codeword in linear time and is robust against any number of tandem duplications of the same length k. Thus, for any two codewords $\mathbf{u}$ and $\mathbf{u}'$ from different subcodes, they are always distinguishable from each other no matter how many tandem duplications of length k affect them. Thus, in order to study the decoding/reconstruction problem for tandem duplications, it suffices to consider the corresponding problem for the code of $\mathcal{C}_{\mathbf{x}}$.

Under certain mapping $\psi_{\mathbf{x}}$, each subcode $\mathcal{C}_{\mathbf{x}}$ can be embedded into the simplex $\Delta_{r(\mathbf{x})}^{m(\mathbf{x})+1}$, where $m(\mathbf{x})$ and $r(\mathbf{x})$ are some integers determined by $\mathbf{x}$, and

$$\Delta_r^m \triangleq \left\{ \mathbf{x} = (x_1, x_2, \ldots, x_{m+1}) \in \mathbb{N}^{m+1} \,\middle|\, \sum_{i=1}^{m+1} x_i = r \right\}.$$

A tandem duplication of length k in $\mathbf{u}$ corresponds to an addition of a unit vector $\mathbf{e}_j \in \mathbb{N}^{m(x)+1}$ to $\psi_{\mathbf{x}}(\mathbf{u})$. Thus, in order to design a reconstruction code with maximum intersection less than N, it is required that for any two distinct codewords $\mathbf{u}, \mathbf{u}' \in \mathcal{C}_{\mathbf{x}}$, it always holds that $d_{\ell_1}(\psi_{\mathbf{x}}(\mathbf{u}), \psi_{\mathbf{x}}(\mathbf{u}')) \geqslant 2\delta$, where δ is the minimum integer such that $\binom{t-\delta+m(\mathbf{x})}{m(\mathbf{x})} < N$. For more details on the code construction and its relation to constant-weight integer codes in the Manhattan metric, the reader may refer to [31, Section III]. Note that the notation N in this section represents the number of reads, while the same notation in [31, Section III] represents the designed size of maximum intersection.

For a vector $\mathbf{x} \in \mathbb{N}^{m+1}$, let

$$\mathcal{B}_t^+(\mathbf{x}) \triangleq \left\{ \mathbf{y} \in \mathbb{N}^{m+1} \,\middle|\, y_i \geqslant x_i \text{ for all } 1 \leqslant i \leqslant m+1, \text{ and } \sum_{i=1}^{m+1} (y_i - x_i) \leqslant t \right\}.$$

Then the reconstruction problem for the code in [31, Section III] can be reduced as follows: Given a code $\mathcal{C} \subseteq \Delta_r^{m+1}$ and a set of vectors $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N\}$ such that $\mathcal{Y} \subseteq \mathcal{B}_t^+(\mathbf{x})$ for some $\mathbf{x} \in \mathcal{C}$, we would like to reconstruct $\mathbf{x}$ from $\mathcal{Y}$. To this end, we use the same decoding process as Algorithm 1. Let $\mathbf{z} = (z_1, z_2, \ldots, z_{m+1})$ where $z_i = \min\{\mathbf{y}_1[i], \mathbf{z}_y[i], \ldots, \mathbf{y}_N[i]\}$ for each $i \in [1, m+1]$. It is easy to see that $\mathbf{x}[i] \leqslant z_i$ for each $i \in [1, m+1]$. Thus, $\mathbf{z} \in \mathcal{B}_r^+(\mathbf{x})$ for some integer r. If $r \geqslant \delta$, then there exist some positions $i_1, i_2, \ldots, i_\tau$ and some positive integers $\delta_1, \delta_2, \ldots, \delta_\tau$ such that $\sum_{j=1}^{\tau} \delta_j = \delta$ and $\mathbf{y}_\ell[i_j] - \mathbf{x}[i_j] \geqslant \delta_j$ for each $j \in [1, \tau]$ and $\ell \in [1, N]$. However, since $N > \binom{m+t-\delta}{m}$, it is impossible. Hence, $r < \delta$ and we may run the decoding algorithm of $\mathcal{C}$ on $\mathbf{z}$ to recover $\mathbf{x}$.

VIII. Conclusion

In this paper, we studied reconstruction and list-reconstruction schemes for integer vectors that suffer from limited-magnitude errors. In Section III we characterized the asymptotic size of the maximum intersection of error balls in relation to the code's minimum distance. Similar problems have been researched for substitutions, insertions and deletions: Levenshtein [17] determined the exact size of the maximum intersection of substitution balls around two words at a given Hamming distance d. Sala *et al.* [20] gave an exact formula for the maximum number of common supersequences shared by sequences at a certain edit distance. Recently, Pham *et al.* [19] provided an asymptotically exact solution for codes which can correct deletions. Interestingly, for all of these four types of errors, the maximum intersection is always $\Theta(n^{f-1})$, where f is the number of excessive errors that the code cannot cope with. We note that the lower and upper bounds presented in Section III differ by a multiplicative factor. It would be interesting to narrow this gap down.

In Section IV we designed two classes of reconstruction codes which both have densities significantly larger than that of the normal error-correcting codes, at the cost of requiring one or two more received sequences. Our codes only deal with a single limited-magnitude error. The design of reconstruction codes against multiple limited-magnitude errors remains unsolved.

We also studied the design of algorithms for the reconstruction problem. We presented two efficient algorithms to reconstruct a transmitted codeword from any given code, and modified our reconstruction algorithms to accommodate the requirement of list decoding when the number of received sequences is less than the maximum intersection. Additionally, we showed that one of our reconstruction algorithm could be used in the context of tandem duplications.

REFERENCES

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

[2] S. Buzaglo and T. Etzion, "Tilings with n-dimensional chairs and their applications to asymmetric codes," *IEEE Trans. Inf. Theory*, vol. 59, no. 3, pp. 1573–1582, Mar. 2013.

[3] K. Cai, H. M. Kiah, T. T. Nguyen, and E. Yaakobi, "Coding for sequence reconstruction for single edits," *IEEE Trans. Inf. Theory*, vol. 68, no. 1, pp. 66–79, Jan. 2022.

[4] Y. Cassuto, M. Schwartz, V. Bohossian, and J. Bruck, "Codes for asymmetric limited-magnitude errors with applications to multilevel flash memories," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1582–1595, Apr. 2010.

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

[6] V. Guruswami, J. Hastad, and S. Kopparty, "On the list-decodability of random linear codes," in *Proc. 42nd Annu. ACM Symp. Theory Comput.*, Cambridge MA, USA, Jun. 2010, pp. 409–416.

[7] W. Hamaker and S. Stein, "Combinatorial packings of R^3 by certain error spheres," *IEEE Trans. Inf. Theory*, vol. 30, no. 2, pp. 364–368, Mar. 1984.

[8] D. Hickerson and S. Stein, "Abelian groups and packing by semicrosses," *Pacific J. Math.*, vol. 122, no. 1, pp. 95–109, 1986.

[9] S. Jain, F. Farnoud, M. Schwartz, and J. Bruck, "Coding for optimized writing rate in DNA storage," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Los Angeles, CA, USA, Jun. 2020, pp. 711–716.

[10] V. Junnila, T. Laihonen, and T. Lehtilä, "On Levenshtein's channel and list size in information retrieval," *IEEE Trans. Inf. Theory*, vol. 67, no. 6, pp. 3322–3341, Jun. 2021.

[11] T. Kløve, J. Luo, I. Naydenova, and S. Yari, "Some codes correcting asymmetric errors of limited magnitude," *IEEE Trans. Inf. Theory*, vol. 57, no. 11, pp. 7459–7472, Nov. 2011.

[12] E. Konstantinova, V. Levenshtein, and J. Siemons, "Reconstruction of permutations distorted by single transposition errors," 2007, *arXiv:math/0702191*.

[13] A. V. Kuznetsov and A. J. H. Vinck, "A coding scheme for single peak-shift correction in (d,k)-constrained channels," *IEEE Trans. Inf. Theory*, vol. 39, no. 4, pp. 1440–1450, Jul. 1993.

[14] V. I. Levenshtein, E. Konstantinova, E. Konstantinov, and S. Molodtsov, "Reconstruction of a graph from 2-vicinities of its vertices," *Discrete Appl. Math.*, vol. 156, pp. 1399–1406, May 2008.

[15] V. I. Levenshtein and J. Siemons, "Error graphs and the reconstruction of elements in groups," *J. Combinat. Theory, A*, vol. 116, no. 4, pp. 795–815, 2009.

[16] V. I. Levenshtein and A. J. H. Vinck, "Perfect (d,k)-codes capable of correcting single peak-shifts," *IEEE Trans. Inf. Theory*, vol. 39, no. 2, pp. 656–662, Mar. 1993.

[17] V. I. Levenshtein, "Efficient reconstruction of sequences," *IEEE Trans. Inf. Theory*, vol. 47, no. 1, pp. 2–22, Jan. 2001.

[18] V. I. Levenshtein, "Efficient reconstruction of sequences from their subsequences or supersequences," *J. Combinat. Theory, A*, vol. 93, no. 2, pp. 310–332, Feb. 2001.

[19] V. Long Phuoc Pham, K. Goyal, and H. Mao Kiah, "Sequence reconstruction problem for deletion channels: A complete asymptotic solution," 2021, *arXiv:2111.04255*.

[20] F. Sala, R. Gabrys, C. Schoeny, and L. Dolecek, "Exact reconstruction from insertions in synchronization codes," *IEEE Trans. Inf. Theory*, vol. 63, no. 4, pp. 2428–2445, Apr. 2017.

[21] M. Schwartz, "Quasi-cross lattice tilings with applications to flash memory," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 2397–2405, Apr. 2012.

[22] M. Schwartz, "On the non-existence of lattice tilings by quasi-crosses," *Eur. J. Combinat.*, vol. 36, pp. 130–142, Feb. 2014.

[23] S. Stein, "Packings of$R^n$by certain error spheres," *IEEE Trans. Inf. Theory*, vol. 30, no. 2, pp. 356–363, Mar. 1984.

[24] S. Stein, "The notched cube tiles R^n," *Discrete Math.*, vol. 80, no. 3, pp. 335–337, 1990.

[25] S. Stein and S. Szabó, *Algebra Tiling*. Washington, DC, USA: Mathematical Association of America, 1994.

[26] H. Wei, X. Wang, and M. Schwartz, "On lattice packings and coverings of asymmetric limited-magnitude balls," *IEEE Trans. Inf. Theory*, vol. 67, no. 8, pp. 5104–5115, Aug. 2021.

[27] H. Wei and M. Schwartz, "Improved coding over sets for DNA-based data storage," *IEEE Trans. Inf. Theory*, vol. 68, no. 1, pp. 118–129, Jan. 2022.

[28] E. Yaakobi and J. Bruck, "On the uncertainty of information retrieval in associative memories," *IEEE Trans. Inf. Theory*, vol. 65, no. 4, pp. 2155–2165, Apr. 2019.

[29] S. Yari, T. Kløve, and B. Bose, "Some codes correcting unbalanced errors of limited magnitude for flash memories," *IEEE Trans. Inf. Theory*, vol. 59, no. 11, pp. 7278–7287, Nov. 2013.

[30] Z. Ye, T. Zhang, X. Zhang, and G. Ge, "Some new results on splitter sets," *IEEE Trans. Inf. Theory*, vol. 66, no. 5, pp. 2765–2776, May 2020.

[31] Y. Yehezkeally and M. Schwartz, "Reconstruction codes for DNA sequences with uniform tandem-duplication errors," *IEEE Trans. Inf. Theory*, vol. 66, no. 5, pp. 2658–2668, May 2020.

[32] Y. Yehezkeally and M. Schwartz, "Uncertainty of reconstruction with list-decoding from uniform-tandem-duplication noise," *IEEE Trans. Inf. Theory*, vol. 67, no. 7, pp. 4276–4287, Jul. 2021.

[33] T. Zhang and G. Ge, "New results on codes correcting single error of limited magnitude for flash memory," *IEEE Trans. Inf. Theory*, vol. 62, no. 8, pp. 4494–4500, Aug. 2016.

[34] T. Zhang and G. Ge, "On the nonexistence of perfect splitter sets," *IEEE Trans. Inf. Theory*, vol. 64, no. 10, pp. 6561–6566, Oct. 2018.

[35] T. Zhang, X. Zhang, and G. Ge, "Splitter sets and K-radius sequences," *IEEE Trans. Inf. Theory*, vol. 63, no. 12, pp. 7633–7645, Dec. 2017.

Hengjia Wei received the Ph.D. degree in applied mathematics from Zhejiang University, Hangzhou, China, in 2014.

He was a Post-Doctoral Fellow with Capital Normal University, Beijing, China, from 2014 to 2016; a Research Fellow with the School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore, from 2016 to 2019; and a Post-Doctoral Fellow with the School of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Israel, from 2019 to 2022. He is currently an Associate Researcher with the Peng Cheng Laboratory, Shenzhen, China. His research interests include combinatorial design theory, coding theory and their intersections. He received the 2017 Kirkman Medal from the Institute of Combinatorics and its Applications.

Moshe Schwartz (Senior Member, IEEE) received the B.A. *(summa cum laude)*, M.Sc., and Ph.D. degrees from the Computer Science Department, Technion—Israel Institute of Technology, Haifa, Israel, in 1997, 1998, and 2004, respectively.

He was a Fulbright Post-Doctoral Researcher with the Department of Electrical and Computer Engineering, University of California at San Diego, and a Post-Doctoral Researcher with the Department of Electrical Engineering, California Institute of Technology. While on sabbatical 2012–2014, he was a Visiting Scientist at the Massachusetts Institute of Technology (MIT). He is currently a Professor with the School of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Israel. His research interests include algebraic coding, combinatorial structures, and digital sequences.

Prof. Schwartz received the 2009 IEEE Communications Society Best Paper Award in Signal Processing and Coding for Data Storage and the 2020 NVMW Persistent Impact Prize. He served as an Associate Editor of coding techniques and coding theory for the IEEE TRANSACTIONS ON INFORMATION THEORY (2014–2021), and since 2021, he has been serving as an Area Editor of coding and decoding for the IEEE TRANSACTIONS ON INFORMATION THEORY. He is also an Editorial Board Member for the *Journal of Combinatorial Theory: Series A* since 2021.