Asymmetric Lee Distance Codes: New Bounds and Constructions

Ryan Gabrys, Han Mao Kiah, and Olgica Milenkovic,
Coordinated Science Laboratory, University of Illinois, Urbana-Champaign
gabrys@illinois.edu, hmkiah@illinois.edu, milenkov@illinois.edu

***Abstract*—We continue our study of a new family of asymmetric Lee codes that arise in the design and implementation of emerging DNA-based storage systems and systems which use parallel string transmission protocols. The codewords are defined over a quaternary alphabet, although the results carry over to other alphabet sizes, and have symbol distances dictated by their underlying binary representation. Our contributions include deriving new bounds for the size of the largest code in this metric based on Delsarte-like linear programming methods and describing new constructions for non-linear asymmetric Lee codes.**

Keywords. Coding for DNA-based storage, Coding theory

I. Introduction

Codes for classical channels with single-sequence inputs and single-sequence outputs have been studied extensively, leading to a diverse suite of schemes such as algebraic codes [16], codes on graphs – e.g., LDPC codes [15] – and polar codes [17]. Similar advances have been reported for parallel channels [8], with the rather common underlying assumption that the channels introduce uncorrelated errors. The alphabet size of the codes used in both scenarios is restricted by the system design, and often, input sequences are de-interleaved or represented as arrays over smaller alphabets in order to enable more efficient transmission. Far less is known about channels that operate on several sequences at the same time and introduce correlated symbol errors, or output reshuffling errors. The goal of this work is to analyze one such scenario, motivated by emerging applications in DNA-based storage systems.

To motivate our analysis, consider a transmission model in which two binary input sequences are simultaneously passed through *two channels* that introduce substitution errors (Figure 1). Simultaneous errors in both strings are less likely than individual string errors. In addition to the substitution errors, the outputs of the channels may be switched – in other words, the label of the channel from which the output symbol originated may be in error. The confusion graph for this type of channel is depicted in Figure 2, where the vertices are indexed by pairs of bits denoting the inputs into the two channels. The labels of the edges denote the channel confusion parameters (weights, distances). More precisely, the parameter $\lambda > 0$ is used to describe the likelihood of certain channel errors.

One application of the aforementioned model arises in DNA sequencing for archival storage [6], where multiple sequences are read in parallel. The readout errors are rare and it is very uncommon to make simultaneous mistakes in both sequences; nevertheless, the identity of the strands may be confused due to string sorting issues. Another unrelated way to view this model is to assume that the binary encodings represent four symbols of the DNA alphabet $\{A, T, G, C\}$, say $00 \to G$, $11 \to C$, $01 \to A$ and $10 \to T$. In this case, the proposed graph describes a DNA readout channel in which the bases $\{A, T\}$ are very likely to be mutually confused during sequencing, while the bases $\{G, C\}$ are much less likely to be misinterpreted for each other. Illumina systems and some other sequencing devices have substitution errors that exhibit such "bias" phenomena, and similar effects may be expected for single base sequencing technologies of the third generation [14].

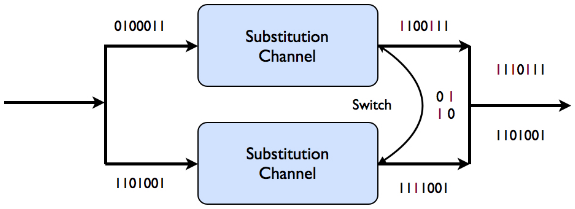

Fig. 1. A pair of channels with individual substitution errors, for which the outputs may also be switched. For the given example, the outputs of the channels at position three are switched.

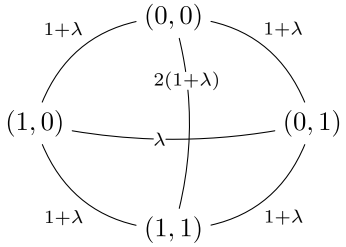

Fig. 2. Weighted confusion graph for codelength $n = 1$.

The problem of interest is to design pairs of sequences – henceforth, termed codewords – such that any two codewords are at a sufficiently large "distance" from each other. For reasons that will become apparent from our later discussion, we subsequently refer to the distance induced by Figure 2 as the *asymmetric Lee distance* (ALD). The ALD resembles a weighted version of the Lee metric [2], with the exception of *two symbols* being treated differently. These two symbols capture the uncertainty about the actual ordering of the readouts. Given the connection with the Lee metric, one may expect ALD code construction questions to be addressed by invoking results for codes in the Lee metric [2]. Nevertheless, due to the asymmetry of the distance, specialized techniques need to be developed to find bounds on the code size and to construct codes that approach these bounds. To accomplish this task, we formally define the ALD as a judiciously chosen convex combination of the Lee and the Hamming distance.

The contributions of this paper are upper bounds on the sizes of codes under the ALD; and non-linear code constructions, which improve upon our results from [5].

II. Problem Formulation

For a positive real number λ, define the ALD distance $d_\lambda((\boldsymbol{a},\boldsymbol{b});(\boldsymbol{c},\boldsymbol{d}))$ between two pairs of sequences as

$$= \sum_{i=1}^{n} (1+\lambda)\left(\chi(a_i,b_i)+\chi(c_i,d_i)\right) + \lambda\chi(a_i,\bar{b}_i,\bar{c}_i,d_i) \quad (1)$$
$$- 2(1+\lambda)\chi(a_i,b_i,c_i,d_i),$$

where $\chi(\cdot,\cdot)$ denotes the standard binary indicator function. It is tedious, but straightforward to verify that $d_\lambda((\boldsymbol{a},\boldsymbol{b});(\boldsymbol{c},\boldsymbol{d}))$ is a metric. For $n=1$, this metric is illustrated in Figure 2.

We henceforth focus our attention on integer-valued λ. For shorthand, we refer to an error which causes a symbol to transition between the states 01 and 10 as a Class I error. We refer to an error which causes a single bit in a symbol to err as a Class II error. Furthermore, we call an error which causes a symbol to transition between 11 and 00 as a Class III error.

Let $\mathcal{C} \in \mathbb{F}_2^n \times \mathbb{F}_2^n$ be such that for any $\boldsymbol{x},\boldsymbol{y} \in \mathcal{C}$, $\boldsymbol{x} \neq \boldsymbol{y}$, one has $d_\lambda(\boldsymbol{x},\boldsymbol{y}) \geqslant d$. For simplicity, we write $d_\lambda(\mathcal{C}) \geqslant d$.

III. Upper Bounds from Delsarte-Like Inequalities

As will be described shortly, in order to capture the distance properties of the codes, we write an element in $\mathbb{Z}_2^n \times \mathbb{Z}_2^n$ as $\boldsymbol{ab}$ and an element in $\mathbb{Z}_2^n \times \mathbb{Z}_2^n \times \mathbb{Z}_2^n$ as $\boldsymbol{abc}$.

Consider the mapping $\phi : \mathbb{Z}_2 \times \mathbb{Z}_2 \to \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2$, which may be simply described by

$$00 \mapsto 000, \qquad 01 \mapsto 010, \qquad 10 \mapsto 100, \qquad 11 \mapsto 111.$$

This map is illustrated in Figure 3. For any positive integer $n > 1$, we extend the definition of the mapping according to $\phi : \mathbb{Z}_2^n \times \mathbb{Z}_2^n \to \mathbb{Z}_2^n \times \mathbb{Z}_2^n \times \mathbb{Z}_2^n$ such that $\phi(\boldsymbol{ab}) = (\phi(a_ib_i))_{i=1}^n$.

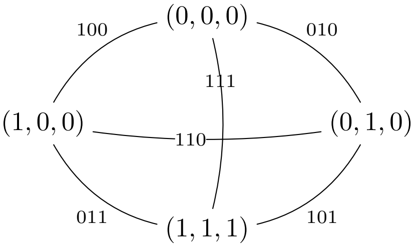

Fig. 3. Confusion graph for extended ternary encoding. Note that the modulo two sum of the label of two vertices equals the label of the edge they define.

Let $\boldsymbol{abc} \in \mathbb{Z}_2^n \times \mathbb{Z}_2^n \times \mathbb{Z}_2^n$. Define the profile of $\boldsymbol{abc}$, denoted by $P(\boldsymbol{abc})$, as $P(\boldsymbol{abc}) = (m_{abc})_{abc \in \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2}$, where $m_{abc} = |\{i : a_ib_ic_i = abc\}|$. With each $abc \in \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2$, associate an indeterminate z_{abc}. Given a collection of words $\mathcal{C} \subseteq \mathbb{Z}_2^n \times \mathbb{Z}_2^n$, define the *complete distance enumerator* of $\mathcal{C}$ as

$$W_{\mathcal{C}}(z_{000}, z_{001}, \ldots, z_{111}) =$$
$$\sum w[m_{000}, m_{001}, \ldots, m_{111}] z_{000}^{m_{000}} z_{001}^{m_{001}} \cdots z_{111}^{m_{111}},$$

where

$$w[m_{000}, m_{001}, \ldots, m_{111}] = \frac{1}{|\mathcal{C}|}$$
$$\left|\{\boldsymbol{ab}, \boldsymbol{pq} \in \mathcal{C} : P(\phi(\boldsymbol{ab}) + \phi(\boldsymbol{pq})) = (m_{000}, m_{001}, \ldots, m_{111})\}\right|,$$

with the sum taken modulo two, and applied component-wise.

Claim 1 *Let $\mathcal{C} \subseteq \mathbb{Z}_2^n \times \mathbb{Z}_2^n$ with $d_\lambda(\mathcal{C}) \geqslant d$. Let $\sum w[m_{000}, \ldots, m_{111}] z_{000}^{m_{000}} \cdots z_{111}^{m_{111}}$ be the complete distance enumerator. Then the following statements are true:*

(i) $|\mathcal{C}| = \sum w[m_{000}, \ldots, m_{111}]$, *which states that the number of codewords may be retrieved by setting all variables to one within the complete weight enumerator;*

(ii) $w[n, 0, \ldots, 0] = 1$, *which follows from the definition of the complete weight enumerator;*

(iii) $w[m_{000}, \ldots, m_{111}] = 0$ *whenever* $m_{001} > 0$, *which ensures that the modulo two sum of vertex labels* 001 *is not encountered;*

(iv) $w[m_{000}, \ldots, m_{111}] = 0$ *if* $(1+\lambda)(m_{010} + m_{100} + m_{101} + m_{011}) + \lambda m_{110} + 2(1+\lambda)m_{111} < d$, *which captures the minimum distance constraint.*

Define the character map

$$\chi(\boldsymbol{pqr}, \boldsymbol{abc}) = (-1)^{\sum_{i=1}^n (p_i+q_i+r_i)(a_i+b_i+c_i)},$$

and observe that $\chi(\boldsymbol{pqr}, \boldsymbol{abc}) = \prod_{i=1}^n \chi(p_iq_ir_i, a_ib_ic_i)$. For $\boldsymbol{pqr}$, suppose that $P(\boldsymbol{pqr}) = (m_{000}, \ldots, m_{111})$ and define $\boldsymbol{z}(\boldsymbol{pqr}) = \prod_{pqr} z_{pqr}^{m_{pqr}}$.

Lemma 1. *Fix $\boldsymbol{abc}$ and suppose $P(\boldsymbol{abc}) = (m_{000}, \ldots, m_{111})$. Then*

$$\sum_{\boldsymbol{pqr}} \boldsymbol{z}(\boldsymbol{pqr}) \chi(\boldsymbol{pqr}, \boldsymbol{abc}) = F(m_{000}, m_{001}, \ldots, m_{111}),$$

where

$$F(m_{000}, m_{001}, \ldots, m_{111})$$
$$= (z_{000} + z_{001} + z_{010} + z_{011} + z_{100} + z_{101} + z_{110} + z_{111})^{m_{000}+m_{011}+m_{101}+m_{110}}$$
$$\times (z_{000} + z_{011} + z_{101} + z_{110} - z_{001} - z_{010} - z_{100} - z_{111})^{m_{001}+m_{010}+m_{100}+m_{111}}.$$

Remark: Observe that $F(m_{000}, m_{001}, \ldots, m_{111})$ consists of two terms, one in which the underlying variables are summed up, and another, in which all variables indexed by vectors of even weight appear with the coefficient $+1$, while variables indexed by vectors of odd weight appear with the coefficient -1. Furthermore, the first term has an exponent equal to the sum of the m-coefficients indexed by vectors of odd weight, while the second term has an exponent equal to the sum of the m-coefficients indexed by vectors of even weight.

Proof: We prove the claimed result by induction. The case $n = 1$ can be verified easily. For $n \geqslant 2$, it suffices to observe that

$$\sum_{p_1p_2q_1q_2r_1r_2} \boldsymbol{z}(p_1p_2q_1q_2r_1r_2)\chi(p_1p_2q_1q_2r_1r_2, a_1a_2b_1b_2c_1c_2) =$$
$$\Big(\sum_{p_1q_1r_1} \boldsymbol{z}(p_1q_1r_1)\chi(p_1q_1r_1, a_1b_1c_1)\Big) \cdot$$
$$\Big(\sum_{p_2q_2r_2} \boldsymbol{z}(p_2q_2r_2)\chi(p_2q_2r_2, a_2b_2c_2)\Big).$$

■

Theorem 2. *Let $\mathcal{C} \subseteq \mathbb{Z}_2^n \times \mathbb{Z}_2^n$. Suppose that $\sum w[m_{000}, \ldots, m_{111}] z_{000}^{m_{000}} \cdots z_{111}^{m_{111}}$ is the complete distance enumerator.*

Define $b[m_{000},\dots,m_{111}]$ *via*

$$\sum w[m_{000},\dots,m_{111}]F(m_{000},\dots,m_{111}) = \\ \sum b[m_{000},\dots,m_{111}]z_{000}^{m_{000}}\cdots z_{111}^{m_{111}}.$$

Then $b[m_{000},\dots,m_{111}] \geqslant 0$, *for all* $m_{000},\dots,m_{111}$.

Proof: Consider the following expression:

$$\sum_{\boldsymbol{ab},\boldsymbol{cd}\in\mathcal{C}} \sum_{\boldsymbol{pqr}} \boldsymbol{z}(\boldsymbol{pqr})\chi(\boldsymbol{pqr},\phi(\boldsymbol{ab})+\phi(\boldsymbol{cd})).$$

On one hand, this expression is given by

$$|\mathcal{C}|\sum w[m_{000},\dots,m_{111}]F(m_{000},\dots,m_{111}).$$

On the other hand, by switching the order of summation, one arrives at

$$\begin{aligned}&\sum_{\boldsymbol{pqr}} \boldsymbol{z}(\boldsymbol{pqr}) \sum_{\boldsymbol{ab},\boldsymbol{cd}\in\mathcal{C}} \chi(\boldsymbol{pqr},\phi(\boldsymbol{ab})+\phi(\boldsymbol{cd}))\\ &= \sum_{\boldsymbol{pqr}} \boldsymbol{z}(\boldsymbol{pqr}) \sum_{\boldsymbol{ab},\boldsymbol{cd}\in\mathcal{C}} \chi(\boldsymbol{pqr},\phi(\boldsymbol{ab}))\chi(\boldsymbol{pqr},\phi(\boldsymbol{cd}))\\ &= \sum_{\boldsymbol{pqr}} \boldsymbol{z}(\boldsymbol{pqr}) \left(\sum_{\boldsymbol{ab}\in\mathcal{C}} \chi(\boldsymbol{pqr},\phi(\boldsymbol{ab}))\right)\left(\sum_{\boldsymbol{cd}\in\mathcal{C}} \chi(\boldsymbol{pqr},\phi(\boldsymbol{cd}))\right)\\ &= \sum_{\boldsymbol{pqr}} \boldsymbol{z}(\boldsymbol{pqr}) \left(\sum_{\boldsymbol{ab}\in\mathcal{C}} \chi(\boldsymbol{pqr},\phi(\boldsymbol{ab}))\right)^2.\end{aligned}$$

The proof follows from $\left(\sum_{\boldsymbol{ab}\in\mathcal{C}} \chi(\boldsymbol{pqr},\phi(\boldsymbol{ab}))\right)^2 \geqslant 0$. ■

As a consequence of the above results, an upper bound for $|\mathcal{C}|$ where $d_\lambda(\mathcal{C})$ is given by the following linear program:

$$\begin{aligned}\text{maximize} \sum w[m_{000},\dots,m_{111}], &\text{subject to}\\ w[n,0,\dots,0] &= 1,\\ w[m_{000},\dots,m_{111}] &= 0, \text{ if } m_{001} > 0,\\ w[m_{000},\dots,m_{111}] &= 0, \text{ if } (1+\lambda)(m_{010}+m_{100}\\ &\quad + m_{101}+m_{011}) + \lambda m_{110}\\ &\quad + 2(1+\lambda)m_{111} < d,\\ \sum w[m_{000},\dots,m_{111}]F(m_{000},\dots,m_{111}) &\geqslant 0.\end{aligned}$$

IV. Computational Results

Observe from Table IV that due to the binary encoding format used for mapping the symbols, the bounds on the size of the codes are powers of two. Furthermore, as can be seen from Tables I and II, the bounds obtained from Delsarte's method are tighter than bounds obtained via the linear programming technique of [10], which we adapted for ALDs in [5]. The only two entries where the Delsarte bounds underperform compared to [5] are highlighted with bold case letters.

V. Code Constructions

In what follows, we present constructions of non-linear codes under the ALD. In many instances, the new code constructions result in codebooks larger than their linear counterparts in [5]. We first analyze ALDs with minimum distance equal to three, and for which $\lambda = 1$. For this parameter case, we improve upon the construction in [5] whenever the block length is $n = 2^v - 2$, for any positive integer v.

We begin by restating a result from [5]. For a positive integer v, let $H_3' \in \mathbb{F}_2^{v\times(2^v-2)}$ be a matrix which has as its columns the non-zero vectors from $\mathbb{F}_2^v$ excluding the all-ones vector. The matrix H_3' is structured so that no two columns are repeated. Write $H_3' = (\boldsymbol{h}_1', \boldsymbol{h}_2', \dots, \boldsymbol{h}_{2^v-2}')$, where for $i \in \{1,\dots,2^v-2\}$, $\boldsymbol{h}_i'$ represents the i-th column of H_3'. Furthermore, let $\boldsymbol{1}_v \in \mathbb{F}_2^{v\times 1}$ be the all-ones vector. Let $\mathcal{C}'(2^v-2,3) \subseteq \mathbb{F}_2^{2^v-2}\times\mathbb{F}_2^{2^v-2}$ be equal to

$$\mathcal{C}'(2^v-2,3) := \Big\{(\boldsymbol{a},\boldsymbol{b}) \in \mathbb{F}_2^{2^v-2}\times\mathbb{F}_2^{2^v-2} : \\ \sum_{i=1}^{2^v-2} a_i\cdot\boldsymbol{h}_i' + \sum_{i=1}^{2^v-2} b_i\cdot\boldsymbol{1}_v = \boldsymbol{0}\Big\}.$$

It has been shown in [5] that $\mathcal{C}'(2^v-2,3)$ has minimum ALD equal to three and the code size equals $|\mathcal{C}'(n,3)| = \frac{4^n}{n+2}$.

We introduce next another code family $\mathcal{C}(n,3)$, where $n = 2^v - 2$ and $v \geqslant 5$, which has minimum ALD three, and $|\mathcal{C}(n,3)| \geqslant \frac{4^n}{n+2} + 2^n$. Note that a code with minimum ALD equal to three, and with $\lambda = 1$, can either: 1) Correct a single Class I error; or 2) Detect a single Class II error. We next describe the code construction and demonstrate that it leads to minimum ALD at least equal to three by showing it can correct either a Class I error or detect a Class II error.

For a vector $\boldsymbol{v} = (\boldsymbol{a},\boldsymbol{b}) = (a_1,\dots,a_n)\times(b_1,\dots,b_n) \in \mathbb{F}_2^n\times\mathbb{F}_2^n$, where $n = 2^v-2$, let $w(\boldsymbol{v}) = |\{i : a_i \neq b_i\}|$. Let $Sup((\boldsymbol{a},\boldsymbol{b})) = (a_{j_1},\dots,a_{j_{w(\boldsymbol{a},\boldsymbol{b})}})$ be the set of all coordinates where the vectors $\boldsymbol{a},\boldsymbol{b}$ disagree, i.e., let $j_1 < j_2 < \cdots < j_{w(\boldsymbol{a},\boldsymbol{b})}$ be the largest collection of integers for which $a_{j_1} \neq b_{j_1}, a_{j_2} \neq b_{j_2}, \dots, a_{j_{w(\boldsymbol{a},\boldsymbol{b})}} \neq b_{j_{w(\boldsymbol{a},\boldsymbol{b})}}$. The basic idea behind our construction is to use two single-error correcting codebooks $\mathcal{S}_1$ and $\mathcal{S}_2$ over the support of vectors in $\mathbb{F}_2^n\times\mathbb{F}_2^n$, where for every $\boldsymbol{v}\in\mathcal{S}_1$, we have $w(\boldsymbol{v}) \leqslant 7$ and $w(\boldsymbol{v}) \equiv 1 \bmod 2$, and for every $\boldsymbol{v}'\in\mathcal{S}_2$, $w(\boldsymbol{v}') \geqslant 9$. Note that for the chosen code property, one cannot have $w(\boldsymbol{v}) = 8$.

Let $\mathcal{C}_H(n,3)_i \subseteq \mathbb{F}_2^i$ denote a binary code of length i with minimum Hamming distance 3. Define

$$\mathcal{C}(n,3)_L := \Big\{(\boldsymbol{a},\boldsymbol{b}) \in \mathcal{S}_1 : Sup((\boldsymbol{a},\boldsymbol{b})) \in \mathcal{C}_H(n,3)_{w(\boldsymbol{a},\boldsymbol{b})}\Big\},$$

$$\mathcal{C}(n,3)_U := \Big\{(\boldsymbol{a},\boldsymbol{b}) \in \mathcal{S}_2 : (\boldsymbol{a},\boldsymbol{b}) \in \mathcal{C}'(n,3)\Big\},$$

Let $\mathcal{C}(n,3) \subseteq \mathbb{F}_2^n\times\mathbb{F}_2^n$ be equal to

$$\mathcal{C}(n,3) := \mathcal{C}(n,3)_L \cup \mathcal{C}(n,3)_U.$$

Lemma 3. *For any positive integer* v, $d_1(\mathcal{C}(2^v-2,3)) \geqslant 3$.

Proof: Let $n = 2^v - 2$. We show that $\mathcal{C}(n,3)$ has minimum ALD 3 by demonstrating that $\mathcal{C}(n,3)$ can either correct a single Class I error or detect a single Class II error. We start by establishing that $\mathcal{C}(n,3)$ can correct a single Class I error. Suppose that $(\boldsymbol{a},\boldsymbol{b}) \in \mathcal{C}(n,3)$ was transmitted and that the vector $(\boldsymbol{c},\boldsymbol{d}) \in \mathbb{F}_2^n\times\mathbb{F}_2^n$ was received, where $(\boldsymbol{c},\boldsymbol{d})$ is the result of a single Class I error occurring in $(\boldsymbol{a},\boldsymbol{b})$ at some position $j \in Supp((\boldsymbol{a},\boldsymbol{b}))$. Notice that Class I errors do not change the locations of the symbols with values in $\{01,10\}$; hence the vector $Supp((\boldsymbol{c},\boldsymbol{d}))$ is the result of a single Class I error occurring

$n\backslash d$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	–	–	2	2																
2	–	–	–	–	4	4	2	2												
3	–	–	–	–	–	–	8	8	4	4	2	2								
4	–	–	–	–	–	–	–	–	16	16	8	8	2	2	2	2				
5	–	–	–	–	–	–	–	–	–	–	**32**	**32**	**16**	**16**	4	4	2	2	2	2

TABLE I
RESULTS OF THE DELSARTE LINEAR PROGRAMMING APPROACH.

$n\backslash d$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	–	–	3	3																
2	–	–	–	–	5	5	5	5												
3	–	–	–	–	–	–	10	10	4	4	4	4								
4	–	–	–	–	–	–	–	–	16	16	9	9	4	4	4	4				
5	–	–	–	–	–	–	–	–	–	–	22	22	10	10	8	8	5	5	4	4

TABLE II
THE RESULTS OF [5].

in $Supp((\boldsymbol{a},\boldsymbol{b}))$. As a result, if $w(\boldsymbol{c},\boldsymbol{d}) = i \leqslant 7$, then we can use the decoder for $\mathcal{C}_H(n,3)_{w(\boldsymbol{a},\boldsymbol{b})}$ to correct the Class I error. Otherwise, if $i \geqslant 9$, we can use the decoder for $\mathcal{C}'(n,3)$ to correct the Class I error.

Suppose that $(\boldsymbol{a},\boldsymbol{b}) \in \mathcal{C}(n,3)$ was transmitted and that the vector $(\boldsymbol{c},\boldsymbol{d}) \in \mathbb{F}_2^n \times \mathbb{F}_2^n$ was received, where $(\boldsymbol{c},\boldsymbol{d})$ is the result of at most one single Class II error occurring in $(\boldsymbol{a},\boldsymbol{b})$. Assume first that $w(\boldsymbol{a},\boldsymbol{b}) = i \leqslant 8$. Since $w(\boldsymbol{a},\boldsymbol{b}) \equiv 1 \bmod 2$, we have $w(\boldsymbol{c},\boldsymbol{d}) \equiv 0 \bmod 2$ if a Class II error occurred, and $w(\boldsymbol{c},\boldsymbol{d}) \equiv 1 \bmod 2$ otherwise. Thus, we can detect a Class II error whenever $w(\boldsymbol{a},\boldsymbol{b}) = i \leqslant 8$. If $w(\boldsymbol{a},\boldsymbol{b}) \geqslant 9$, the result follows from the arguments we presented in [5]. ■

The next lemma provides a lower bound on the cardinality of $\mathcal{C}(n,3)$. Due to space limitations, some derivations in the proof are omitted.

Lemma 4. *For* $v \geqslant 5$, $|\mathcal{C}(2^v-2,3)| \geqslant \frac{4^n}{n+2} + 2^n$.

Proof: We first show that $\mathcal{C}(n,3) \geqslant \mathcal{C}'(n,3)$. Using an averaging argument, one can prove that

$$|\mathcal{C}(n,3)_L| \geqslant 2^n \cdot \sum_{k=1, k \text{ odd}}^{7} \frac{\binom{n}{k}}{8}.$$

Furthermore,

$$|\mathcal{C}(n,3)_U| \geqslant 2^n \cdot \sum_{k=9}^{n} \frac{\binom{n}{k}}{n+2}, \text{ so that}$$

$$\begin{aligned} |\mathcal{C}(n,3)| \geqslant & 2^n \cdot \left(\sum_{k=1, k \text{ odd}}^{7} \frac{\binom{n}{k}}{8} + \sum_{k=9}^{n} \frac{\binom{n}{k}}{n+2} \right) \\ \geqslant & 2^n \cdot \frac{\sum_{k=1, k \text{ odd}}^{7} \frac{n+2}{8} \binom{n}{k} + \sum_{k=9}^{n} \binom{n}{k}}{n+2} \end{aligned}$$

From [5], we also have

$$|\mathcal{C}'(n,3)| = \frac{4^n}{n+2} = \frac{2^n \cdot \sum_{k=0}^{n} \binom{n}{k}}{n+2}.$$

We show next that $|\mathcal{C}(n,3)| - |\mathcal{C}'(n,3)| \geqslant 0$, which implies

$$\frac{2^n \cdot G(n)}{n+2} \geqslant 0,$$

where

$$G(n) = \frac{n+2}{8} \sum_{k=1, k \text{ odd}}^{7} \binom{n}{k} - \sum_{k=0}^{8} \binom{n}{k}.$$

Consider the following ratio

$$\begin{aligned} \frac{\sum_{k=0}^{8} \binom{n}{k}}{\frac{n+2}{8} \sum_{k=1, k \text{ odd}}^{7} \binom{n}{k}} &= \frac{8 \sum_{k=0}^{8} \binom{n}{k}}{(n+2) \sum_{k=0}^{7} \binom{n-1}{k}} \\ &= \frac{8n \sum_{k=0}^{8} \frac{1}{(n-k)!k!}}{(n+2) \sum_{k=0}^{7} \frac{1}{(n-k-1)!k!}} n. \end{aligned}$$

Clearly, $G(n) \geqslant 0$ if the quantity above is less than one. It is straightforward, but tedious, to show that

$$\frac{8n \sum_{k=0}^{8} \frac{1}{(n-k)!k!}}{(n+2) \sum_{k=0}^{7} \frac{1}{(n-k-1)!k!}} \leqslant 1$$

for $n \geqslant 30$, implying $G(n) \geqslant 0$ and $|\mathcal{C}(n,3)| \geqslant |\mathcal{C}'(n,3)|$.

Notice that we can strengthen the lower bound on $|\mathcal{C}(n,3)|$,

$$\begin{aligned} |\mathcal{C}(n,3)| \geqslant 2^n \cdot \Big(& \frac{\binom{n}{1} + \binom{n}{3}}{4} + \frac{\binom{n}{5} + \binom{n}{7}}{8} \\ & + \sum_{k=9}^{n} \frac{1}{n+2} \binom{n}{k} \Big) \\ = & 2^n \cdot \left(\sum_{k=1, k \text{ odd}}^{7} \frac{\binom{n}{k}}{8} + \sum_{k=9}^{n} \frac{\binom{n}{k}}{n+2} \right) + \\ & 2^n \cdot \left(\frac{\binom{n}{1} + \binom{n}{3}}{8} \right) \geqslant \frac{4^n}{n+2} + 2^n, \end{aligned}$$

as claimed. ■

We consider next the case $d \geqslant 3$ and $\lambda = 1$. The basic idea will be to map the symbols from $\mathbb{F}_2 \times \mathbb{F}_2$ to $\{0,1,2,3\}$ and then use codes in the Lee metric. The map $\phi : \mathbb{F}_2 \times \mathbb{F}_2 \to \{0, 1, 2, 3\}$ of interest is defined as follows:

$$\phi(0,0) \to 0, \phi(0,1) \to 1, \phi(1,0) \to 2, \phi(1,1) \to 3.$$

For a vector $(\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{F}_2^n \times \mathbb{F}_2^n$, let $\phi(\boldsymbol{a}, \boldsymbol{b}) = (\phi(a_1, b_1), \ldots, \phi(a_n, b_n))$. The image of a set under the map ϕ is the set of images of elements of the set under ϕ.

Let p be an odd prime, and suppose that v is a positive integer. Let $u \in \mathbb{F}_d$ and $\boldsymbol{z} \in \mathbb{F}_{p^v}^{\lfloor \frac{d}{2} \rfloor}$. Furthermore, suppose that α is a primitive element of $\mathbb{F}_{p^v}$. Then, for $n = p^v - 1$, define $\mathcal{C}_1(n, d, u, \boldsymbol{z}) \subseteq \mathbb{F}_2^n \times \mathbb{F}_2^n$ as

$$\begin{aligned}\mathcal{C}_1(n, d, u, \boldsymbol{z}) := \Big\{ & (\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{F}_2^n \times \mathbb{F}_2^n : \\ & \sum_{i=1}^{n} \phi(a_i, b_i) \equiv u \mod d \\ & \sum_{i=1}^{n} \phi(a_i, b_i) \cdot \alpha^1 = z_1 \\ & \vdots \\ & \sum_{i=1}^{n} \phi(a_i, b_i) \cdot \alpha^{\lfloor d/2 \rfloor} = z_{\lfloor d/2 \rfloor} \},\end{aligned} \tag{2}$$

where all the operations are over $\mathbb{F}_{p^v}$.

Lemma 5. *For $n = p^v - 1$, $u \in \mathbb{F}_d$ and $\boldsymbol{z} \in \mathbb{F}_{p^v}^{\lfloor \frac{d}{2} \rfloor}$, $d_1(\mathcal{C}_1(n, d, u, \boldsymbol{z})) \geqslant d$.*

Proof: Suppose that $(\boldsymbol{a}, \boldsymbol{b}), (\boldsymbol{c}, \boldsymbol{d}) \in \mathcal{C}_1(n, d, u, \boldsymbol{z})$. Let $I = |\{m : a_m + b_m = c_m + d_m = 1, a_m \neq c_m\}|$, $J = |\{m : a_m + b_m = c_m + d_m = 2, a_m \neq c_m\}|$, and $K = |\{m : (a_m, b_m) \neq (c_m, d_m)\}| - (I + J)$. If $d_1((\boldsymbol{a}, \boldsymbol{b}), (\boldsymbol{c}, \boldsymbol{d})) \geqslant d$, then we need to show $I + 2K + 4J \geqslant d$. Clearly, if the Lee distance of $\phi(\mathcal{C}_1(n, d, u, \boldsymbol{z}))$ is at least d, then $I + 2K + 3J \geqslant d$ and the result holds.

To see that the Lee distance of $\mathcal{C}_1(n, d, u, \boldsymbol{z})$ is at least d, notice that we can recover any error vector of weight at most $\lfloor \frac{d}{2} \rfloor$ from the power sums listed in (2). More specifically if the Lee weight of $\phi(\boldsymbol{c}, \boldsymbol{d})$ is at most $\lfloor \frac{d}{2} \rfloor$, it is known from [16] that given $\sum_{i=1}^{n} \phi(c_i, d_i) \cdot \alpha^k$ for $k = \{0, 1, \ldots, \lfloor \frac{d}{2} \rfloor\}$, we can uniquely determine the vector $\phi(\boldsymbol{c}, \boldsymbol{d})$. Clearly, from (2), we have the information on $\sum_{i=1}^{n} \phi(c_i, d_i) \cdot \alpha^k$ for $k \in \{1, \ldots, \lfloor \frac{d}{2} \rfloor\}$. For $k = 0$, we can uniquely determine $\sum_{i=1}^{n} \phi(c_i, d_i)$ as by assumption, the Lee weight of $\phi(\boldsymbol{c}, \boldsymbol{d})$ is at most $\lfloor \frac{d}{2} \rfloor$. ■

Using an averaging argument, we have

$$|\mathcal{C}_1(n, d, u, \boldsymbol{z})| \geqslant \frac{4^n}{d(n+1)^{\lfloor d/2 \rfloor}},$$

which for $d \geqslant 7$ improves the lower bounds in [5].

The next construction can be used to generate codes under the ALD for a larger range of parameters. A full analysis of the underlying methodology is deferred to the full version of the manuscript.

Let $\mathcal{C}_L(n, d)$ denote a code over $\mathbb{F}_3$ of length n with minimum Lee distance d. Similarly, let $\mathcal{C}_H(n, d)_\ell$ denote a binary code of length ℓ and minimum Hamming distance d. Define $\mathcal{C}_\lambda(n, d) \subseteq \mathbb{F}_2^n \times \mathbb{F}_2^n$ as:

$$\begin{aligned}\mathcal{C}_\lambda(n, d) :=& \{(\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{F}_2^n \times \mathbb{F}_2^n : \\ & (a_1 + b_1, a_2 + b_2, \ldots, a_n + b_n) \in \mathcal{C}_L(n, \lceil \frac{d}{1+\lambda} \rceil), \\ & Supp((\boldsymbol{a}, \boldsymbol{b})) \in \mathcal{C}_H(n, \lceil \frac{d}{\lambda} \rceil)_{Supp((\boldsymbol{a}, \boldsymbol{b}))}\},\end{aligned} \tag{3}$$

where for $i \in \{1, \ldots, n\}$, $a_i + b_i = 0$ if $a_i = b_i = 0$, $a_i + b_i = 2$ if $a_i = b_i = 1$, and $a_i + b_i = 1$ otherwise.

We have the following lemma.

Lemma 6. *Given n and d, one has $d_\lambda(\mathcal{C}_\lambda(n, d)) \geqslant d$.*

Proof: Suppose $(\boldsymbol{a}, \boldsymbol{b}), (\boldsymbol{c}, \boldsymbol{d}) \in \mathcal{C}_\lambda(n, d)$. Let $I = |\{m : a_m + b_m = c_m + d_m = 1, a_m \neq c_m\}|$, $J = |\{m : a_m + b_m = c_m + d_m = 2, a_m \neq c_m\}|$, and $K = |\{m : (a_m, b_m) \neq (c_m, d_m)\}| - (I + J)$. Recall that if $d_\lambda((\boldsymbol{a}, \boldsymbol{b}), (\boldsymbol{c}, \boldsymbol{d})) \geqslant d$, we need to show that $\lambda \cdot I + (1 + \lambda) \cdot K + 2(1 + \lambda) \cdot J \geqslant d$. If $(a_1 + b_1, \ldots, a_m + b_m) \neq (c_1 + d_1, \ldots, c_m + d_m)$, then $(1 + \lambda) \cdot K + 2(1 + \lambda) \cdot J \geqslant (1 + \lambda) \cdot (K + 2J)$, and since $K + 2J \geqslant \lceil \frac{d}{1+\lambda} \rceil$, the result follows from (3). Otherwise, if $(a_1 + b_1, \ldots, a_m + b_m) = (c_1 + d_1, \ldots, c_m + d_m)$, the result is an immediate consequence of (3). ■

Acknowledgment: This work was funded by the Strategic Research Initiative (SRI) program at the University of Illinois and by the NSF STC on Science of Information.

References

[1] H. Astola and I. Tabus, "Bounds on the Size of Lee-Codes," *International Symposium on Image and Signal Processing and Analysis*, Sept. 2013.

[2] J. Astola, "The theory of Lee-codes," *Lappeenranta University of Technology, Department of Physics and Mathematics, Research Report*, Jan. 1982.

[3] G. M. Church, Y. Gao, and S. Kosuri, "Next-generation digital information storage in DNA," *Science*, vol. 337, no. 6102, pp. 1628-1628, Sep. 2012.

[4] R. Feynman, "There's plenty of room at the bottom," Caltech, Pasadena. 29 Dec. 1959. Lecture.

[5] R. Gabrys, H.M. Kiah, O. Milenkovic, "Asymmetric Lee Distance Codes for DNA-Based Storage," submitted to ISIT, 2015.

[6] N. Goldman et al., "Towards practical, high-capacity, low-maintenance information storage in synthesized DNA," *Nature*, Jan. 2013.

[7] J. I. Hall, *Notes on Coding Theory*, available at http://www.mth.msu.edu/∼jhall/classes/codenotes/coding-notes.html, 2013.

[8] E. Hof et al., "Capacity-achieving polar codes for arbitrarily permuted parallel channels," *IEEE Trans. on Info. Theory*, vol. 59, no. 3, pp. 1505-1516, March 2013.

[9] K.A.S. Immink, *Codes for mass data storage systems*, Shannon Foundation Publisher, 2004.

[10] A.A. Kulkarni, and N. Kiyavash, "Nonasymptotic upper bounds for deletion correcting codes," *IEEE Trans. on Info. Theory*, vol. 59, no. 8, pp. 5115-5130, April 2013.

[11] V.I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet physics doklady*, vol. 10, pp. 707-710, 1966.

[12] A. Mazumdar, A. Barg, and N. Kashyap, "Coding for high-density recording on a 1-D granular magnetic medium," *IEEE Trans. on Info. Theory*, vol. 57, no. 11, pp. 7403-7417, June 2011.

[13] O. Milenkovic and N. Kashyap, "On the design of codes for DNA computing," *Coding and Cryptography*, Springer Berlin Heidelberg, pp. 100-119, 2006.

[14] K. Nakamura, et al., "Sequence-specific error profile of Illumina sequencers," *Nucleic acids research*, voll. 39, no. 13, 2011.

[15] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. on Info. Theory*, vol. 47, no. 2, pp. 599-618, August 2002.

[16] Roth, R., *Introduction to coding theory*, Cambridge University Press, 2006.

[17] Tal, I., Vardy, A., "How to construct polar codes," *IEEE Trans. on Info. Theory*, vol. 59, no. 10, pp. 6562 - 6582, September 2013.

[18] Welch, L.R., McEliece, R.J., Rumsey, H,. "A low-rate improvement on the Elias bound," *IEEE Trans. Inform. Theory*, vol. 20, no. 5, pp. 676-678, 6 Jan. 2003.