Equivalence of Insertion/Deletion Correcting Codes for d-dimensional Arrays

Evagoras Stylianou, **Lorenz Welter**, **Rawad Bitar**, **Antonia Wachter-Zeh**, and **Eitan Yaakobi**

***Abstract*— We consider the problem of correcting insertion and deletion errors in the d-dimensional space. This problem is well understood for vectors (one-dimensional space) and was recently studied for arrays (two-dimensional space). For vectors and arrays, the problem is motivated by several practical applications such as DNA-based storage and racetrack memories. From a theoretical perspective, it is interesting to know whether the same properties of insertion/deletion correcting codes generalize to the d-dimensional space. In this work, we show that the equivalence between insertion and deletion correcting codes generalizes to the d-dimensional space. As a particular result, we show the following missing equivalence for arrays: a code that can correct t_{r} and t_{c} row/column deletions can correct any combination of $t_{\mathrm{r}}^{\mathrm{ins}}+t_{\mathrm{r}}^{\mathrm{del}}=t_{\mathrm{r}}$ and $t_{\mathrm{c}}^{\mathrm{ins}}+t_{\mathrm{c}}^{\mathrm{del}}=t_{\mathrm{c}}$ row/column insertions and deletions. The fundamental limit on the redundancy and a construction of insertion/deletion correcting codes in the d-dimensional space remain open for future work.**

I. Introduction

Coding for insertions and deletions received a lot of attention due to new applications such as DNA-based data storage [1], [2], synchronization errors [3], [4] and racetrack memories [5]. An important notion in this class of codes is the equivalence of insertion and deletion errors. In his original work [6], Levenshtein showed that a code can correct t deletions in a length-n vector if and only if it can correct any combination of t_{i} insertions and t_{d} deletions such that $t_{\mathrm{i}}+t_{\mathrm{d}}=t$. A more intuitive proof of the equivalence, which line of thoughts we follow in this work, is given in [7]. A code $\mathcal{C}$ correcting deletions in q-ary length-n vectors is evaluated by its redundancy defined as $R \triangleq n-\log_q|\mathcal{C}|$. The redundancy of t-deletion-correcting codes is bounded from below by $t\log_q n-\mathcal{O}(1)$ [6], [7]. The asymptotical tightness of this bound is shown using the Varshamov-Tenengolts codes [6], [8], [9] that can correct one deletion. Several recent works considered constructing binary t-deletion-correcting codes, $t>1$, whose redundancy approach the previously mentioned lower bound [10]–[17].

Codes correcting insertions and deletions in two-dimensional arrays have been investigated in [18]–[23]. The model considered in [20]–[23] is that of coding for row/column insertions and deletions in two-dimensional arrays. In [22], Hagiwara constructed codes that can correct up to t_c column and t_r row deletions where t_r and t_c are predetermined. In [20], [23], the authors constructed codes correcting a variable number of column and row deletions for a predetermined number of total deletions. In addition, they provided a lower bound on the redundancy of codes correcting insertions and deletions in arrays. Moreover, they generalized the equivalence between insertions and deletions across each dimension (columns and rows), separately. More precisely, the authors showed that given an integer t, a code can correct t_r and t_c, for all $t_r+t_c=t$, row and column deletions if and only if it can correct the same number of rows/columns insertions. However, combinations of insertions and deletions of columns (and rows) was not studied.

In this work we generalize the equivalence between codes correcting insertions and deletions to the d-dimensional space. In this setting, the insertions and deletions are defined as $(d-1)$-dimensional hyperplane insertions/deletions in a d-dimensional array. In the d-dimensional space there are $d=\binom{d}{d-1}$ different types of $(d-1)$-hyperplane deletions/insertions. Each type of deletion is indexed by the missing dimension. More precisely, let $(x_1,\ldots,x_d)$ describe the axes of the d-dimensional space. Deleting a $(d-1)$-dimensional hyperplane not containing the axis x_i is referred to as an x_i-deletion. See Fig. 1 for an illustrative example for $d=3$. For a vector $\boldsymbol{t}=(t_1,\ldots,t_d)$, a $\boldsymbol{t}$-deletion refers to the combination of t_i x_i-deletions for $i\in\{1,\ldots,d\}$. We show that a code can correct $\boldsymbol{t}$-deletions if and only if it can correct $\boldsymbol{t}$-insertions. We extend this result to combinations of insertion and deletions, i.e., we show that a code can correct $\boldsymbol{t}$-deletions if and only if it can correct any combination of $\boldsymbol{t}^{\mathrm{del}}$-deletions and $\boldsymbol{t}^{\mathrm{ins}}$-insertions such that $\boldsymbol{t}^{\mathrm{del}}+\boldsymbol{t}^{\mathrm{ins}}=\boldsymbol{t}$. We show that the number of x_i-errors (insertions plus deletions) must remain the same for the equivalence to hold.

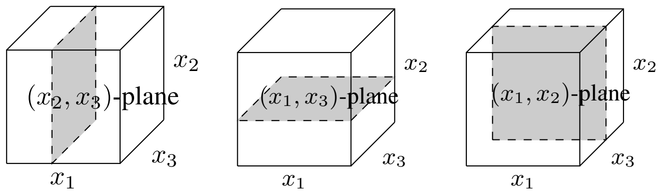
II. Notation and Preliminaries

Denote the q-ary alphabet by $\Sigma_q \triangleq \{0,\ldots,q-1\}$ and the set of integers $\{1,\ldots,n\}$ by $[n]$. Moreover, denote the set of d-dimensional arrays, in short called arrays, by $\Sigma_q^{\bigotimes_{i=1}^d n_i} = \Sigma_q^{n_1\times\cdots\times n_d} \triangleq \Sigma_q^{n_1}\times\cdots\times\Sigma_q^{n_d}$ with entries in Σ_q. We abbreviate $\Sigma_q^{n^{\otimes d}} \triangleq \Sigma_q^{\bigotimes_{i=1}^d n}$, if $n_i=n_j$ for all $i,j\in[d]$. Let $(x_1,\ldots,x_d)$ describe the axes of the d-dimensional space. We call an x_k-deletion in an array $\mathbf{X}\in\Sigma_q^{n^{\otimes d}}$ the deletion of a $(d-1)$-dimensional hyperplane spanned only along the axes $(x_1,\ldots,x_{k-1},x_{k+1},\ldots,x_d)$. For two-dimensional arrays an

ES, LW, RB and AW-Z are with the ECE department at the Technical University of Munich. EY is with the CS department of Technion — Israel Institute of Technology. Emails: {evagoras.stylianou, lorenz.welter, rawad.bitar, antonia.wachter-zeh}@tum.de, yaakobi@cs.technion.ac.il.

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 801434), from the Technical University of Munich - Institute for Advanced Studies, funded by the German Excellence Initiative and European Union Seventh Framework Programme under Grant Agreement No. 291763.

978-1-6654-2159-1/22/$31.00 ©2022 IEEE



(a) x_1-deletion (b) x_2-deletion (c) x_3-deletion

Fig. 1: Illustration of all possible plane deletion or insertion in a 3-dimensional array.

x_1-deletion corresponds to a column deletion and an x_2-deletion to a row deletion. See Fig. 1 for an illustration in the 3-dimensional space. A $\boldsymbol{t}$-deletion where $\boldsymbol{t} \in \mathbb{Z}_{\geq 0}^d$ corresponds to the combination of t_i x_i-deletions for $i \in [d]$, resulting in an array $\widetilde{\mathbf{X}} \in \Sigma_q^{\bigotimes_{i=1}^d (n-t_i)}$. Moreover, a $\boldsymbol{t}$-insdel where $\boldsymbol{t} = \boldsymbol{t}^{\text{ins}} + \boldsymbol{t}^{\text{del}}$ corresponds to the combination of $\boldsymbol{t}^{\text{del}}$-deletions and $\boldsymbol{t}^{\text{ins}}$-insertions resulting in an array $\widetilde{\mathbf{X}} \in \Sigma_q^{\bigotimes_{i=1}^d (n+(t_i^{\text{ins}}-t_i^{\text{del}}))}$.

For $\mathbf{X} \in \Sigma_q^{n^{\otimes d}}$ and $\boldsymbol{t}^{(d)} \in \mathbb{Z}_{\geq 0}^d$, the set of arrays resulting from a $\boldsymbol{t}^{(d)}$-deletion in $\mathbf{X}$ is called the deletion "ball" and is denoted by $\mathbb{D}_{\boldsymbol{t}}^d(\mathbf{X})$. We define a $\boldsymbol{t}^{(d)}$-deletion correcting code $\mathcal{C} \subseteq \Sigma_q^{n^{\otimes d}}$ as the code that can correct any $\boldsymbol{t}^{(d)}$-deletion for all $\mathbf{X} \in \mathcal{C}$. The all-zero vector with "1" in the i-th position is denoted by $\boldsymbol{e}_i$. The $\mathbf{1}^{(d)}$ denotes the all-one vector of length d. Vectors $\boldsymbol{t}$ of the form $\boldsymbol{t} = (t, \ldots, t)$ are denoted by $t\mathbf{1}^{(d)}$. For such vectors we denote the deletion ball by $\mathbb{D}_{t\mathbf{1}}^d(\mathbf{X})$. The $\boldsymbol{t}$-insertion and the insertion balls $\mathbb{I}_{\boldsymbol{t}}^d(\mathbf{X})$ and $\mathbb{I}_{t\mathbf{1}}^d(\mathbf{X})$ are defined similarly. Moreover, the set of arrays resulting from $\boldsymbol{t}^{(d)}$-insdel in $\mathbf{X}$ is called the insertion-deletion "ball" and denoted by $\mathbb{ID}_{\boldsymbol{t}}^d(\mathbf{X})$. We define a $\boldsymbol{t}^{(d)}$-insdel correcting code $\mathcal{C} \subseteq \Sigma_q^{n^{\otimes d}}$ as the code that can correct any $\boldsymbol{t}^{(d)}$-insdel for all $\mathbf{X} \in \mathcal{C}$. For an integer $t \geq 0$, a $t^{(d)}$-deletion refers to the collection of all possible $\boldsymbol{t}$-deletions such that $\sum_{i=1}^d t_i = t$. We define a $t^{(d)}$-deletion correcting code $\mathcal{C} \subseteq \Sigma_q^{n^{\otimes d}}$ as the code that can correct any $t^{(d)}$-deletion for all $\mathbf{X} \in \mathcal{C}$. The same notation is used for insertions. For an integer a, we define $\delta_a(x)$ to be equal to one if $x = a$ and zero otherwise.

For $j \in [d]$, the projection $\mathcal{P}_j$ projects an array $\mathbf{X} \in \Sigma_q^{\bigotimes_{i=1}^d n_i}$ along the x_j-th axis to an array $\mathcal{P}_j(\mathbf{X}) \in \Sigma_{q^{n_j}}^{\bigotimes_{i\in[d]}^{i\neq j} n_i}$. The projection $\mathcal{P}_j$ preserves the order of the axes, i.e., it projects $\mathbf{X}$ from the space with axes $(x_1, \ldots, x_d)$ onto the space with axes $(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_d)$. Moreover, we denote by $\mathcal{P}_j^{-1}$ the inverse projection, or the expansion, of an array $\mathbf{X} \in \Sigma_{q^{n_j}}^{\bigotimes_{i\in[d]}^{i\neq j} n_i}$ along the x_j-th axis to obtain an array $\mathcal{P}_j^{-1}(\mathbf{X}) \in \Sigma_q^{\bigotimes_{i=1}^d n_i}$. The inverse projection $\mathcal{P}_j^{-1}$ also preserves the order of the dimensions $(x_1, \ldots, x_d)$. For example, given an array $\mathbf{X} \in \Sigma_{q^n}^{n\times n}$ in the (x_1, x_3) space, the inverse projection $\mathcal{P}_2^{-1}(\mathbf{X})$ expands each entry of $\mathbf{X}$ to a vector in Σ_q^n along x_2 to obtain $\mathcal{P}_2^{-1}(\mathbf{X}) \in \Sigma_q^{n\times n\times n}$.

Next we state in our notation two preliminary results derived in [20], [23] for the 2-dimensional case. Lemma 1 is used as a building block of our proofs.

Theorem 1. *[23, Theorem 1] A code $\mathcal{C} \subseteq \Sigma_q^{n\times n}$ is a $t^{(2)}$-deletion correcting code if and only if it is a $t^{(2)}$-insertion correcting code, i.e., for any arrays $\mathbf{X}, \mathbf{Y} \in \Sigma_q^{n\times n}$,*

$$\mathbb{D}_{\boldsymbol{t}}^2(\mathbf{X}) \cap \mathbb{D}_{\boldsymbol{t}}^2(\mathbf{Y}) \neq \emptyset \text{ if and only if } \mathbb{I}_{\boldsymbol{t}}^2(\mathbf{X}) \cap \mathbb{I}_{\boldsymbol{t}}^2(\mathbf{Y}) \neq \emptyset.$$

for any choice of $\boldsymbol{t} \in \mathbb{Z}_{\geq 0}^2$ such that $t_1 + t_2 = t$.

Lemma 1. *[20] For a positive integer m and $i, j \in [2]$, any two arrays $\mathbf{X} \in \Sigma_q^{(m+\delta_1(i))\times(m+\delta_2(i))}$ and $\mathbf{Y} \in \Sigma_q^{(m+\delta_1(j))\times(m+\delta_2(j))}$, it holds that*

$$\mathbb{D}_{\boldsymbol{e}_i}^2(\mathbf{X}) \cap \mathbb{D}_{\boldsymbol{e}_j}^2(\mathbf{Y}) \neq \emptyset \Leftrightarrow \mathbb{I}_{\boldsymbol{e}_j}^2(\mathbf{X}) \cap \mathbb{I}_{\boldsymbol{e}_i}^2(\mathbf{Y}) \neq \emptyset.$$

III. SYMMETRIC INSERTION/DELETION EQUIVALENCE

In this section we prove the following theorem.

Theorem 2. *A code $\mathcal{C} \subseteq \Sigma_q^{n^{\otimes d}}$ is a $t\mathbf{1}^{(d)}$-deletion-correcting code if and only if it is a $t\mathbf{1}^{(d)}$-insertion-correcting code.*

To prove Theorem 2 we need three intermediate results. In Claim 1, we show that $\boldsymbol{t}^{(d)}$-deletions and $\boldsymbol{t}^{(d)}$-insertions in an array $\mathbf{X}$ are not affected by the projection $\mathcal{P}_\kappa(\mathbf{X})$ and the inverse projection $\mathcal{P}_\kappa^{-1}(\mathbf{X})$ such that $t_\kappa = 0$. We then extend Lemma 1 to the d-dimensional space, cf., Lemma 2, and use it as a building block in our proofs. In particular, we use Lemma 2 to prove Theorem 3 showing that a code is a $\mathbf{1}^{(d)}$-deletion-correcting code if and only if it is a $\mathbf{1}^{(d)}$-insertion-correcting code. Having the aforementioned results, proving Theorem 2 follows by showing that for any $\mathbf{X}, \mathbf{Y} \in \Sigma_q^{n^{\otimes d}}$, $\mathbb{D}_{t\mathbf{1}}^d(\mathbf{X}) \cap \mathbb{D}_{t\mathbf{1}}^d(\mathbf{Y}) \neq \emptyset$ if and only if $\mathbb{I}_{t\mathbf{1}}^d(\mathbf{X}) \cap \mathbb{I}_{t\mathbf{1}}^d(\mathbf{Y}) \neq \emptyset$. The proof holds by using the exact same steps as in the proof of [20, Corollary 2], but extended to the d-dimensional space and is thus omitted.

We start with the first intermediate result.

Claim 1. *For any two vectors $\boldsymbol{r}_1, \boldsymbol{r}_2 \in \mathbb{N}^d$ such that there exists a $\kappa \in [d]$ for which $r_{1,\kappa} = r_{2,\kappa} = 0$ and any two arrays $\mathbf{X} \in \Sigma_q^{\bigotimes_{i=1}^d (n+r_{1,i})}$, $\mathbf{Y} \in \Sigma_q^{\bigotimes_{i=1}^d (n+r_{2,i})}$, it holds that,*

$$\mathbb{D}_{\boldsymbol{r}_1}^d(\mathbf{X}) \cap \mathbb{D}_{\boldsymbol{r}_2}^d(\mathbf{Y}) \neq \emptyset \Leftrightarrow \mathbb{D}_{\mathcal{P}_\kappa(\boldsymbol{r}_1)}^{d-1}(\mathcal{P}_\kappa(\mathbf{X})) \cap \mathbb{D}_{\mathcal{P}_\kappa(\boldsymbol{r}_2)}^{d-1}(\mathcal{P}_\kappa(\mathbf{Y})) \neq \emptyset,$$

where κ denotes the κ-th dimension in the d-dimensional space and $\mathcal{P}_\kappa(\boldsymbol{r}_j) \in \mathbb{N}^{d-1}$ is equal to $\boldsymbol{r}_j$ with the zero deleted in the κ-th position for $j = 1, 2$.
The same statement holds for the insertion case.

Proof. We first prove the "if" part. Let $\mathbf{D} \in \mathbb{D}_{\boldsymbol{r}_1}^d(\mathbf{X}) \cap \mathbb{D}_{\boldsymbol{r}_2}^d(\mathbf{Y})$ and $\mathbf{D}' \in \mathbb{D}_{\mathcal{P}_\kappa(\boldsymbol{r}_1)}^{d-1}(\mathcal{P}_\kappa(\mathbf{X})) \cap \mathbb{D}_{\mathcal{P}_\kappa(\boldsymbol{r}_2)}^{d-1}(\mathcal{P}_\kappa(\mathbf{Y}))$. The κ-th dimension is not affected by a deletion in both arrays $\mathbf{X}$ and $\mathbf{Y}$. Therefore, the deletions do not affect the mapping of the q-ary symbols to q^n-ary symbols along the axis x_κ, when using the projection function. Thus, the $(d-1)$-dimensional hyperplane deletions in $\mathbf{X}, \mathbf{Y}$ correspond to $(d-2)$-dimensional hyperplane deletions in the respective projected arrays. Hence, we have $\mathcal{P}_\kappa^{-1}(\mathbf{D}') = \mathbf{D}$.

We now prove the "only if" part. By expanding the q^n-ary symbols to q-ary symbols along the x_κ-th axis, i.e., by applying the inverse projection, the $(d-2)$-dimensional hyperplane deletions in $\mathcal{P}_\kappa(\mathbf{X}), \mathcal{P}_\kappa(\mathbf{Y})$ transform to $(d-1)$-dimensional hyperplane deletions in $\mathbf{X}, \mathbf{Y}$ with no x_κ-deletions. This follows from the definition of the projections. □

We now state and prove the second intermediate result.

Lemma 2. *For positive integers* $m_1, \ldots, m_d$ *and* $i, j \in [d]$, *for any two arrays* $\mathbf{X} \in \Sigma_q^{\bigotimes_{\ell=1}^{d}(m_\ell+\delta_\ell(i))}$ *and* $\mathbf{Y} \in \Sigma_q^{\bigotimes_{\ell=1}^{d}(m_\ell+\delta_\ell(j))}$ *it holds that,*

$$\mathbb{D}^d_{\boldsymbol{e}_i}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{e}_j}(\mathbf{Y}) \neq \emptyset \Leftrightarrow \mathbb{I}^d_{\boldsymbol{e}_j}(\mathbf{X}) \cap \mathbb{I}^d_{\boldsymbol{e}_i}(\mathbf{Y}) \neq \emptyset.$$

Proof. We only show the "if" part. The "only if" part is proven similarly. We prove the statement by induction over the dimensions. The two-dimensional case, i.e., $d = 2$, was already shown in [20] and is recalled in Lemma 1. To illustrate the proof techniques used in the proof and in this work, we choose the three-dimensional case as the base case of the induction. Without loss of generality, we show that

$$\mathbb{D}^d_{\boldsymbol{e}_1}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{e}_2}(\mathbf{Y}) \neq \emptyset \Leftrightarrow \mathbb{I}^d_{\boldsymbol{e}_2}(\mathbf{X}) \cap \mathbb{I}^d_{\boldsymbol{e}_1}(\mathbf{Y}) \neq \emptyset.$$

Base case $d = 3$*:* We show that

$$\mathbb{D}^3_{\boldsymbol{e}_1}(\mathbf{X}) \cap \mathbb{D}^3_{\boldsymbol{e}_2}(\mathbf{Y}) \neq \emptyset \Leftrightarrow \mathbb{I}^3_{\boldsymbol{e}_2}(\mathbf{X}) \cap \mathbb{I}^3_{\boldsymbol{e}_1}(\mathbf{Y}) \neq \emptyset.$$

For $\mathbf{X} \in \Sigma_q^{(n+1)\times n \times n}$ and $\mathbf{Y} \in \Sigma_q^{n \times (n+1) \times n}$ let $\mathbf{D} \in \mathbb{D}^3_{\boldsymbol{e}_1}(\mathbf{X}) \cap \mathbb{D}^3_{\boldsymbol{e}_2}(\mathbf{Y})$. Since the deletion does not affect both arrays along the axis x_3, then we can project along this axis to transform the given three-dimensional deletion problem to a two-dimensional deletion problem by Claim 1. Thus, the $\boldsymbol{e}_1$-deletion in $\mathbf{X}$ converts to a row deletion in $\mathcal{P}_3(\mathbf{X})$ and the $\boldsymbol{e}_2$-deletion in $\mathbf{Y}$ to a column deletion in $\mathcal{P}_3(\mathbf{Y})$. Hence, it holds that $\mathcal{P}_3(\mathbf{D}) \in \mathbb{D}^2_{\boldsymbol{e}_1}(\mathcal{P}_3(\mathbf{X})) \cap \mathbb{D}^2_{\boldsymbol{e}_2}(\mathcal{P}_3(\mathbf{Y}))$. By Lemma 1, we have the following statement

$$\begin{aligned}&\mathbb{D}^2_{\boldsymbol{e}_1}(\mathcal{P}_3(\mathbf{X})) \cap \mathbb{D}^2_{\boldsymbol{e}_2}(\mathcal{P}_3(\mathbf{Y})) \neq \emptyset \\ &\Leftrightarrow \mathbb{I}^2_{\boldsymbol{e}_2}(\mathcal{P}_3(\mathbf{X})) \cap \mathbb{I}^2_{\boldsymbol{e}_1}(\mathcal{P}_3(\mathbf{Y})) \neq \emptyset.\end{aligned}$$

Therefore, there exists a $\mathcal{P}_3(\mathbf{I}) \in \mathbb{I}^2_{\boldsymbol{e}_2}(\mathcal{P}_3(\mathbf{X})) \cap \mathbb{I}^2_{\boldsymbol{e}_1}(\mathcal{P}_3(\mathbf{Y}))$. Let $\mathbf{I} = \mathcal{P}_3^{-1}(\mathcal{P}_3(\mathbf{I}))$, by Claim 1 the previous statement is equivalent to stating that there exists a $\mathbf{I} \in \mathbb{I}^3_{\boldsymbol{e}_2}(\mathbf{X}) \cap \mathbb{I}^3_{\boldsymbol{e}_1}(\mathbf{Y})$. This results from applying the inverse projection $\mathcal{P}_3^{-1}(\cdot)$ on the respective arrays, transforming the row/column insertions in the two-dimensional space to $\boldsymbol{e}_1$-/$\boldsymbol{e}_2$-insertion in the three-dimensional space; thus concluding the base case.

Induction hypothesis: For a positive integer $d-1$ assume that it holds that

$$\mathbb{D}^{d-1}_{\boldsymbol{e}_1}(\mathbf{X}) \cap \mathbb{D}^{d-1}_{\boldsymbol{e}_2}(\mathbf{Y}) \neq \emptyset \Leftrightarrow \mathbb{I}^{d-1}_{\boldsymbol{e}_2}(\mathbf{X}) \cap \mathbb{I}^{d-1}_{\boldsymbol{e}_1}(\mathbf{Y}) \neq \emptyset.$$

Induction step: Given the induction hypothesis we show that the equivalence holds also for d, i.e.,

$$\mathbb{D}^d_{\boldsymbol{e}_1}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{e}_2}(\mathbf{Y}) \neq \emptyset \Leftrightarrow \mathbb{I}^d_{\boldsymbol{e}_2}(\mathbf{X}) \cap \mathbb{I}^d_{\boldsymbol{e}_1}(\mathbf{Y}) \neq \emptyset,$$

and let $\mathbf{D} \in \mathbb{D}^d_{\boldsymbol{e}_1}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{e}_2}(\mathbf{Y})$. To apply Claim 1 and use the induction hypothesis, we project the arrays on an axis different than the ones affected by a deletion. For the given case, we have $d-2$ available axes to project on. Assume we project on the axis x_κ, where $\kappa \in [d] \setminus \{1, 2\}$. Thus, we transform the $(d-1)$-dimensional hyperplane deletion in $\mathbf{X}$ and $\mathbf{Y}$ to a $(d-2)$-dimensional hyperplane deletion in $\mathcal{P}_\kappa(\mathbf{X})$ and $\mathcal{P}_\kappa(\mathbf{Y})$ (c.f. Claim 1). Therefore, we can write that

$$\begin{aligned}&\mathbb{D}^d_{\boldsymbol{e}_1}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{e}_2}(\mathbf{Y}) \neq \emptyset, \\ &\Leftrightarrow \mathbb{D}^{d-1}_{\boldsymbol{e}_1}(\mathcal{P}_\kappa(\mathbf{X})) \cap \mathbb{D}^{d-1}_{\boldsymbol{e}_2}(\mathcal{P}_\kappa(\mathbf{Y})) \neq \emptyset, \\ &\Leftrightarrow \mathbb{I}^{d-1}_{\boldsymbol{e}_2}(\mathcal{P}_\kappa(\mathbf{X})) \cap \mathbb{I}^{d-1}_{\boldsymbol{e}_1}(\mathcal{P}_\kappa(\mathbf{Y})) \neq \emptyset,\end{aligned}$$

where the last equivalence follows from the induction hypothesis. Hence, there exists a $\mathcal{P}_\kappa(\mathbf{I}) \in \mathbb{I}^{d-1}_{\boldsymbol{e}_2}(\mathcal{P}_\kappa(\mathbf{X})) \cap \mathbb{I}^{d-1}_{\boldsymbol{e}_1}(\mathcal{P}_\kappa(\mathbf{Y}))$. Due to the fact that we have projected on an axis $x_\kappa \neq x_1, x_2$ and given Claim 1, we can interpret the $(d-2)$-dimensional hyperplane insertion in $\mathcal{P}_\kappa(\mathbf{X})$ and $\mathcal{P}_\kappa(\mathbf{Y})$ as a $(d-1)$-dimensional hyperplane insertion in $\mathbf{X}$ and $\mathbf{Y}$ by applying the inverse projection $\mathcal{P}_\kappa^{-1}(\cdot)$ to the projected arrays. By the above observations we conclude that there exists a $\mathbf{I} \in \mathbb{I}^d_{\boldsymbol{e}_2}(\mathbf{X}) \cap \mathbb{I}^d_{\boldsymbol{e}_1}(\mathbf{Y})$ if there exists $\mathbf{D} \in \mathbb{D}^d_{\boldsymbol{e}_1}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{e}_2}(\mathbf{Y})$ and conclude the "if" part of the proof. ☐

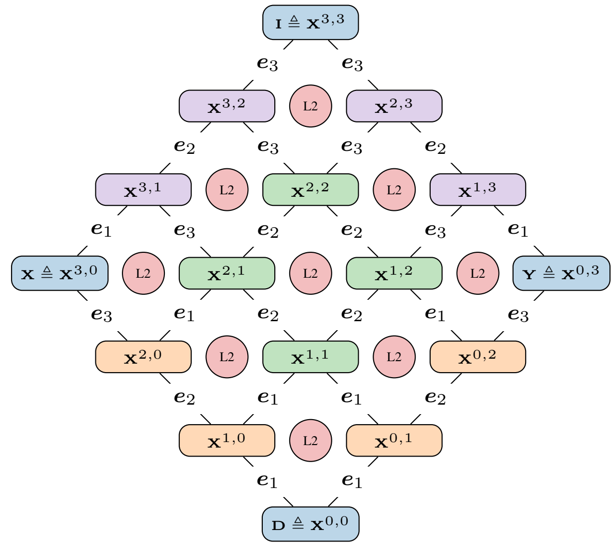


Fig. 2: A flow chart of the proof of Theorem 3 for $d = 3$. *Given a array* $\mathbf{D} \in \mathbb{D}^d_{\mathbf{1}}(\mathbf{X}) \cap \mathbb{D}^d_{\mathbf{1}}(\mathbf{Y})$, *we show the existence* $\mathbf{I} \in \mathbb{I}^d_{\mathbf{1}}(\mathbf{X}) \cap \mathbb{I}^d_{\mathbf{1}}(\mathbf{Y})$. *Given the existence of* $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{D}$, *and the orange arrays one can show by Lemma 2 the existence of the green and purple marked arrays and the array* $\mathbf{I}$. *Since* $\mathbf{X}, \mathbf{Y}$ *and* $\mathbf{I}$ *are connected via the purple arrays as shown one can conclude the equivalence.*

We now show the equivalence of $\mathbf{1}^{(d)}$-insertion and $\mathbf{1}^{(d)}$-deletion-correcting codes by using the results of Claim 1 and Lemma 2.

Theorem 3. *A code* $\mathcal{C} \subseteq \Sigma_q^{n^{\otimes d}}$ *is a* $\mathbf{1}^{(d)}$*-deletion-correcting code if and only if it is a* $\mathbf{1}^{(d)}$*-insertion-correcting code.*

Proof. We provide an illustration of the proof for the case of $d = 3$ in Fig. 2. Assume there exists an array $\mathbf{D} \in \Sigma_q^{(n-1)^{\otimes d}}$ such that $\mathbf{D} \in \mathbb{D}^d_{\mathbf{1}}(\mathbf{X}) \cap \mathbb{D}^d_{\mathbf{1}}(\mathbf{Y})$. For simplicity of notation, we fix the order of the deletions in $\mathbf{X}$ and $\mathbf{Y}$ to obtain $\mathbf{D}$ to be an x_d-deletion first, then an x_{d-1}-deletion and so on until making an x_1-deletion. Note that the proof can be replicated for any ordering by the comprehensiveness of Lemma 2 which is our main building block. To prove the statement, we build a grid-like structure with axes $i, j \in [d]$ and arrays as grid points denoted by $\mathbf{X}^{i,j}$. We define $\mathbf{X}^{d,0} \triangleq \mathbf{X}$, $\mathbf{X}^{0,d} \triangleq \mathbf{Y}$, and $\mathbf{X}^{0,0} \triangleq \mathbf{D}$. For fixed $j = 0$, let the series of arrays $\{\mathbf{X}^{i,0}\}_{i=0}^{d}$ be defined such that $\mathbf{X}^{i-1,0} \in \mathbb{D}^d_{\boldsymbol{e}_i}(\mathbf{X}^{i,0})$ for $i = \{1, \ldots, d\}$. We define the series of arrays $\{\mathbf{X}^{0,j}\}_{j=0}^{d}$ similarly for fixed $i = 0$. The strategy of the proof will show the existence of arrays $\mathbf{X}^{i,j}$ for any $i, j \in [d]$ such that $\mathbf{X}^{d,d} \in \mathbb{I}^d_{\mathbf{1}}(\mathbf{X}^{d,0}) \cap \mathbb{I}^d_{\mathbf{1}}(\mathbf{X}^{0,d})$.

By the definition of the series we have that $\mathbf{X}^{0,0} \in \mathbb{D}^d_{\boldsymbol{e}_1}(\mathbf{X}^{1,0}) \cap \mathbb{D}^d_{\boldsymbol{e}_1}(\mathbf{X}^{0,1})$. By Lemma 2 there exists an array $\mathbf{X}^{1,1} \in \mathbb{I}^d_{\boldsymbol{e}_1}(\mathbf{X}^{1,0}) \cap \mathbb{I}^d_{\boldsymbol{e}_1}(\mathbf{X}^{0,1})$. From that it follows that $\mathbf{X}^{1,0} \in \mathbb{D}^d_{\boldsymbol{e}_2}(\mathbf{X}^{2,0}) \cap \mathbb{D}^d_{\boldsymbol{e}_1}(\mathbf{X}^{1,1})$. By applying again Lemma 2 we have that there exists an array $\mathbf{X}^{2,1} \in \mathbb{I}^d_{\boldsymbol{e}_1}(\mathbf{X}^{2,0}) \cap \mathbb{I}^d_{\boldsymbol{e}_2}(\mathbf{X}^{1,1})$. For $j = 1$, by repeating the aforementioned strategy we can show the existence of the series of arrays $\{\mathbf{X}^{i,1}\}_{i=2}^{d}$. Given this series of arrays one can show the existence $\{\mathbf{X}^{i,2}\}_{i=1}^{d}$, where $j = 2$ and given the starting statement $\mathbf{X}^{0,1} \in \mathbb{D}^d_{\boldsymbol{e}_1}(\mathbf{X}^{1,1}) \cap \mathbb{D}^d_{\boldsymbol{e}_2}(\mathbf{X}^{0,2})$. Therefore by consecutively incrementing $j \in [0, d-1]$ and for each j incrementing consecutively $i \in [0, d-1]$, then for each pair (i, j) by Lemma 2 one has the following equivalence: Given $\mathbf{X}^{i,j} \in \mathbb{D}^d_{\boldsymbol{e}_i}(\mathbf{X}^{i+1,j}) \cap \mathbb{D}^d_{\boldsymbol{e}_j}(\mathbf{X}^{i,j+1})$ there exists an array $\mathbf{X}^{i+1,j+1} \in \mathbb{I}^d_{\boldsymbol{e}_j}(\mathbf{X}^{i+1,j}) \cap \mathbb{I}^d_{\boldsymbol{e}_i}(\mathbf{X}^{i,j+1})$. Therefore, we have proven the existence of an array $\mathbf{X}^{d,d} \in \mathbb{I}^d_{\mathbf{1}}(\mathbf{X}^{d,0}) \cap \mathbb{I}^d_{\mathbf{1}}(\mathbf{X}^{0,d})$ which concludes the proof. $\square$

IV. Equivalence of Insertion and Deletions Correcting Codes: General Case

In this section we show the the equivalence of $\boldsymbol{t}^{(d)}$-insertions and $\boldsymbol{t}^{(d)}$-deletions in d-dimensional arrays for any number of $(d-1)$-dimensional hyperplane insertions and deletions, respectively, i.e., we show the equivalence of $\boldsymbol{t}^{(d)}$-deletion-correcting codes with $\boldsymbol{t}^{(d)}$-insertion-correcting codes for any $\boldsymbol{t}^{(d)} \in \mathbb{Z}^d_{\geq 0}$. The proof follows similar steps as the one used by the authors in [23] for the two-dimensional case.

Theorem 4. *A code $\mathcal{C} \subseteq \Sigma_q^{n^{\otimes d}}$ is a $\boldsymbol{t}^{(d)}$-deletion-correcting code if and only if it is a $\boldsymbol{t}^{(d)}$-insertion-correcting code.*

Proof. For notational convenience we define the vector $\boldsymbol{c}^j \triangleq (c_1, \ldots, c_{j-1}, 0, c_{j+1}, \ldots, c_d) \in \mathbb{N}^d$. In this proof, the vector $\boldsymbol{t}^{(d)}$ can be written as $\boldsymbol{t}^{(d)} = t_j \mathbf{1}^{(d)} + \boldsymbol{c}^j \triangleq \boldsymbol{t}_{j,c}$, where $t_j = \min_{i \in [d]} t_i$ and $c_i \triangleq t_i - t_j$ for $i \in [d]$, to emphasize the composition. Without loss of generality, we show the proof for $j = 1$, since by symmetry the proof holds for all $j \in [d]$, and write $t \triangleq t_1$. Let $t' \triangleq \sum_{i=1}^{d} c_i$, the proof proceeds by induction over t'. For simplicity, we fix in some parts of the proof the order of the x_i-deletions. That serves for a better presentation of the proofs and incurs no loss of generality. In the proof the contraposition is shown, i.e., we show that $\mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y}) \neq \emptyset$ if and only if $\mathbb{I}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X}) \cap \mathbb{I}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y}) \neq \emptyset$. We only show the "if" part since the "only if" part follows by using similar arguments.

Base case $\sum_{i=1}^{d} c_i = 1$: For the reader's convenience, a flowchart of the proof for $d = 3$ is presented in Fig. 3. There are $d-1$ possibilities for $\boldsymbol{c}^1$ such that $\sum_{i=1}^{d} c_i = 1$. We show

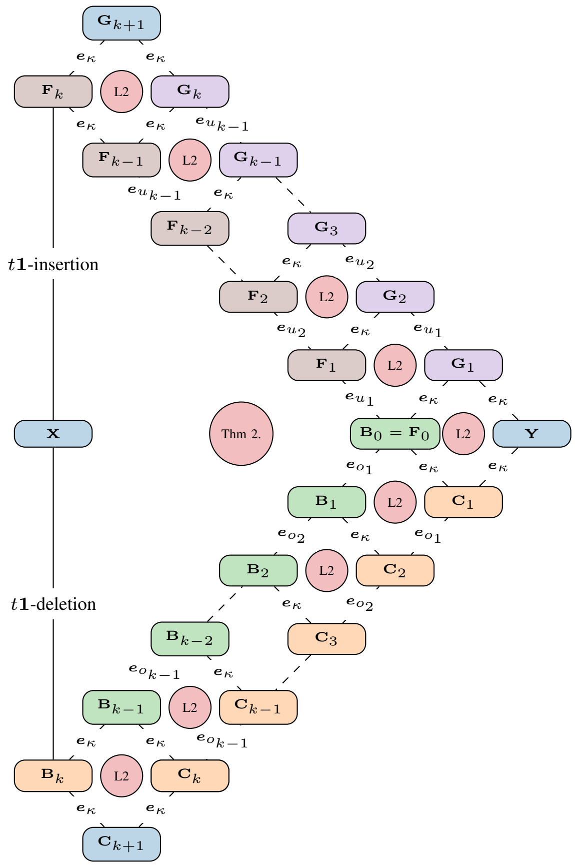


Fig. 3: A flow chart of the proof of Theorem 4 for $d = 3$. Given an array $\mathbf{C}_{k+1} \in \mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y})$, we show the existence $\mathbf{G}_{k+1} \in \mathbb{I}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X}) \cap \mathbb{I}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y})$. Given the existence of $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{C}_{k+1}$ and the orange arrays one can show by Lemma 2 the existence of the green marked arrays and then by Theorem 2 and Lemma 2 the existence of brown and purple marked arraysand the array $\mathbf{G}_{k+1}$.

the proof steps for $\boldsymbol{c}^1 = \boldsymbol{e}_\kappa$, i.e., there is a combination of $t\mathbf{1}^{(d)}$-deletions and an extra x_κ-deletion for $\kappa \in [d]$.

For any two arrays $\mathbf{X}, \mathbf{Y} \in \Sigma_q^{n^{\otimes d}}$, assume there exists a array $\mathbf{D}$ such that $\mathbf{D} \in \mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y})$. Let $k = dt$, define the array $\mathbf{B}_k$ such that $\mathbf{B}_k \in \mathbb{D}^d_{t\mathbf{1}}(\mathbf{X})$ and $\mathbf{D} \in \mathbb{D}^d_{\boldsymbol{e}_\kappa}(\mathbf{B}_k)$ due to the choice of $\boldsymbol{c}^1$. For simplicity, we define $\mathbf{C}_1 \in \mathbb{D}^d_{\boldsymbol{e}_\kappa}(\mathbf{Y})$. Let $\boldsymbol{o} \in [d]^k$ denote the vector whose entries o_i denote the series of x_{o_i}-deletions to obtain $\mathbf{D}$ from $\mathbf{C}_1$ and fix $o_k = \kappa$. We define the series of arrays $\{\mathbf{C}_s\}_{s=1}^{k+1}$ such that

$$\mathbf{C}_s \in \begin{cases} \mathbb{D}^d_{\boldsymbol{e}_\kappa}(\mathbf{C}_{s-1}) & \text{if } s = 1 \text{ or } s = k+1 \\ \mathbb{D}^d_{\boldsymbol{e}_{o_{s-1}}}(\mathbf{C}_{s-1}) & \text{otherwise.} \end{cases}$$

where $\mathbf{C}_0 \triangleq \mathbf{Y}$ and $\mathbf{C}_{k+1} \triangleq \mathbf{D}$. We show that there exists a series of arrays $\{\mathbf{B}_s\}_{s=0}^{k-1}$, resulting from hyperplane insertions starting from $\mathbf{B}_k$ and leading to an array $\mathbf{B}_0 \in \Sigma_q^{n^{\otimes d}}$, such that $\mathbf{B}_k \in \mathbb{D}^d_{t\mathbf{1}}(\mathbf{X}) \cap \mathbb{D}^d_{t\mathbf{1}}(\mathbf{B}_0)$. By the aforementioned definitions we have that $\mathbf{C}_{k+1} \in \mathbb{D}^d_{\boldsymbol{e}_\kappa}(\mathbf{B}_k) \cap \mathbb{D}^d_{\boldsymbol{e}_\kappa}(\mathbf{C}_k)$. By Lemma 2, there exists a $\mathbf{B}_{k-1} \in \mathbb{I}^d_{\boldsymbol{e}_\kappa}(\mathbf{B}_k) \cap \mathbb{I}^d_{\boldsymbol{e}_\kappa}(\mathbf{C}_k)$. Applying Lemma 2 sequentially shows the existence of the series of arrays $\{\mathbf{B}_s\}_{s=0}^{k-1}$, i.e., by Lemma 2 for each $\mathbf{C}_{s+1} \in \mathbb{D}^d_{\boldsymbol{e}_\kappa}(\mathbf{B}_s) \cap \mathbb{D}^d_{\boldsymbol{e}_{o_s}}(\mathbf{C}_s)$

there exists a $\mathbf{B}_{s-1} \in \mathbb{I}_{\boldsymbol{e}_{o_s}}(\mathbf{B}_s) \cap \mathbb{I}^d_{\boldsymbol{e}_\kappa}(\mathbf{C}_s)$ for $s \in \{k, \ldots, 1\}$. Hence, we show the existence of an array $\mathbf{B}_0 \in \Sigma_q^{n^{\otimes d}}$ such that $\mathbf{B}_k \in \mathbb{D}^d_{t\mathbf{1}}(\mathbf{X}) \cap \mathbb{D}^d_{t\mathbf{1}}(\mathbf{B}_0)$.

By Theorem 2, the existence of $\mathbf{B}_k$ implies the existence of an array $\mathbf{F}_k \in \mathbb{I}^d_{t\mathbf{1}}(\mathbf{X}) \cap \mathbb{I}^d_{t\mathbf{1}}(\mathbf{B}_0)$, i.e., obtained by a $t\mathbf{1}^{(d)}$-insertion in $\mathbf{B}_0$. Let $\boldsymbol{u} \in [d]^k$ denote the vector whose entries u_i denote the series of x_{u_i}-insertions to obtain $\mathbf{F}_k$ from $\mathbf{B}_0$ and fix $u_k = \kappa$. We define the arrays $\{\mathbf{F}_s\}_{s=1}^k$ such that

$$\mathbf{F}_s \in \begin{cases} \mathbb{I}^d_{\boldsymbol{e}_\kappa}(\mathbf{F}_{s-1}) & \text{if } s = k \\ \mathbb{I}^d_{\boldsymbol{e}_{u_s}}(\mathbf{F}_{s-1}) & \text{otherwise,} \end{cases}$$

where $\mathbf{F}_0 \triangleq \mathbf{B}_0$. Noting that $\mathbf{C}_1 \in \mathbb{D}^d_{\boldsymbol{e}_\kappa}(\mathbf{B}_0) \cap \mathbb{D}^d_{\boldsymbol{e}_\kappa}(\mathbf{Y})$ and applying Lemma 2, there exists an array $\mathbf{G}_1 \in \mathbb{I}^d_{\boldsymbol{e}_\kappa}(\mathbf{B}_0) \cap \mathbb{I}^d_{\boldsymbol{e}_\kappa}(\mathbf{Y})$, which means that $\mathbf{F}_0 \in \mathbb{D}^d_{\boldsymbol{e}_{u_1}}(\mathbf{F}_1) \cap \mathbb{D}^d_{\boldsymbol{e}_\kappa}(\mathbf{G}_1)$. By sequentially applying Lemma 2 we can show the existence of the series of arrays $\{\mathbf{G}_s\}_{s=1}^{k+1}$ such that $\mathbf{G}_{k+1} \in \mathbb{I}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X}) \cap \mathbb{I}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y})$. Meaning by the fact that $\mathbf{F}_{s-1} \in \mathbb{D}^d_{\boldsymbol{e}_{u_s}}(\mathbf{F}_s) \cap \mathbb{D}^d_{\boldsymbol{e}_\kappa}(\mathbf{G}_s)$ there exists an array $\mathbf{G}_{s+1} \in \mathbb{I}^d_{\boldsymbol{e}_\kappa}(\mathbf{F}_s) \cap \mathbb{I}^d_{\boldsymbol{e}_{u_s}}(\mathbf{G}_s)$ for $s \in \{1, \ldots, k\}$. Hence, we have shown that if there exists an array $\mathbf{C}_{k+1} \in \mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y})$, then there exists an array $\mathbf{G}_{k+1} \in \mathbb{I}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X}) \cap \mathbb{I}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y})$, which concludes the base case.

Induction hypothesis: Given any vector $\boldsymbol{c}^1$ such that $\sum_{i=1}^d c_i = t'$, and two arrays $\mathbf{X}, \mathbf{Y} \in \Sigma_q^{n^{\otimes d}}$ it holds that

$$\mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y}) \neq \emptyset \iff \mathbb{I}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X}) \cap \mathbb{I}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y}) \neq \emptyset,$$

where $\boldsymbol{t}_{1,c} \triangleq t\mathbf{1} + \boldsymbol{c}^1$.

Induction step: Assume that the induction hypothesis holds for all values $0 \leq \sum_{i=1}^d c_i = t'$ where $c_1 = 0$. We prove that the hypothesis holds for $\sum_{i=1}^d c_i + 1 = t' + 1$, i.e., by adding an extra hyperplane deletion. Let the extra deletion be an x_κ-deletion and define $\boldsymbol{t}'_{1,c} = (t, t+c_2, \ldots, t+c_\kappa+1, \ldots, t+c_d)$. Assume that there exists an array $\mathbf{D}$ such that $\mathbf{D} \in \mathbb{D}^d_{\boldsymbol{t}'_{1,c}}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{t}'_{1,c}}(\mathbf{Y})$. Let $k' = dt + t' + 1$, then we defined the arrays $\mathbf{B}_{k'}$ and $\mathbf{C}_{k'}$ such that $\mathbf{B}_{k'} \in \mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{X})$ and $\mathbf{C}_{k'} \in \mathbb{D}^d_{\boldsymbol{t}_{1,c}}(\mathbf{Y})$. The rest of the proof follows from the base case, by using k' instead of k and therefore is omitted due to space limitations. $\square$

By considering the collections of all $\boldsymbol{t}^{(d)}$-deletion-correcting codes such that $\sum_{i=1}^d t_i = t$ we have the following corollary.

Corollary 1. *A code $\mathcal{C} \subseteq \Sigma_q^{n^{\otimes d}}$ is a $t^{(d)}$-deletion-correcting code if and only if it is a $t^{(d)}$-insertion-correcting code.*

V. Insdel Equivalence

So far we have only considered the equivalence between insertion and deletion correcting codes. In this section we are going to discuss the equivalence between $\boldsymbol{t}^{(d)}$-deletion and $\boldsymbol{t}^{(d)}$-insdel correcting codes. First, we need the following claim.

Claim 2. *For positive integers $m_1, \ldots, m_d$, $i \in [d]$, a vector $\boldsymbol{r}^i = (0, \ldots, 0, r_i, 0, \ldots, 0)$, and any two arrays $\mathbf{X}, \mathbf{Y} \in \Sigma_q^{\bigotimes_{\ell=1}^d m_\ell}$ it holds that*

$$\mathbb{D}^d_{\boldsymbol{r}^i}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{r}^i}(\mathbf{Y}) \neq \emptyset \Leftrightarrow \mathbb{ID}^d_{\boldsymbol{r}^i}(\mathbf{X}) \cap \mathbb{ID}^d_{\boldsymbol{r}^i}(\mathbf{Y}) \neq \emptyset.$$

Proof. We only show the "if" part, since the "only if" part follows by similar arguments. Let $\mathbf{D} \in \mathbb{D}^d_{\boldsymbol{r}^i}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{r}^i}(\mathbf{Y})$. We define a consecutive series of projections of an array $\mathbf{X}$ along the axes in a set $\mathcal{I} \subseteq [d]$ by $\mathcal{P}_\mathcal{I}(\mathbf{X})$. Let $\mathcal{I} = [d] \setminus \{i\}$, we have $\mathcal{P}_\mathcal{I}(\mathbf{X}), \mathcal{P}_\mathcal{I}(\mathbf{Y}) \in \Sigma^n_{q^{n(d-1)}}$. Since we do not project along the axis affected by deletions we can transform the $(d-1)$-hyperplane deletions to symbol deletions in $\mathcal{P}_\mathcal{I}(\mathbf{X}), \mathcal{P}_\mathcal{I}(\mathbf{Y})$ by Claim 1. Thus, there exits a $\mathcal{P}_\mathcal{I}(\mathbf{D}) \in \mathbb{D}^1_{r_i}(\mathcal{P}_\mathcal{I}(\mathbf{X})) \cap \mathbb{D}^1_{r_i}(\mathcal{P}_\mathcal{I}(\mathbf{Y}))$ such that $\mathcal{P}_\mathcal{I}^{-1}(\mathcal{P}_\mathcal{I}(\mathbf{D})) = \mathbf{D}$. Hence, by [7] there exists a $\mathcal{P}_\mathcal{I}(\mathbf{I}) \in \mathbb{ID}^1_{r_i}(\mathcal{P}_\mathcal{I}(\mathbf{X})) \cap \mathbb{ID}^1_{r_i}(\mathcal{P}_\mathcal{I}(\mathbf{Y}))$. According to Claim 1 it follows that there exists a $\mathcal{P}_\mathcal{I}^{-1}(\mathcal{P}_\mathcal{I}(\mathbf{I})) = \mathbf{I} \in \mathbb{ID}^d_{\boldsymbol{r}^i}(\mathbf{X}) \cap \mathbb{ID}^d_{\boldsymbol{r}^i}(\mathbf{Y})$, since all entries of $\boldsymbol{r}^i$ are zero except the i-th position. $\square$

It is important to note that the position of r_i within the vector $\boldsymbol{r}^i$ must remain the same for any equivalence. This means that x_i-deletions are only equivalent to x_i-insdels and not to x_j-insdels, $j \neq i$. We show this idea through a counterexample for two-dimensional arrays.

Counterexample 1. *The equivalence of a $(1,0)$-deletion-correcting code and a $(0,1)$-deletion-correcting code does not hold. To show this, we consider two arrays $\mathbf{X}, \mathbf{Y} \in \Sigma^{3\times 3}$ and assume there exists an array $\mathbf{D} \in \Sigma^{2\times 3}$ such that $\mathbf{D} \in \mathbb{D}^2_{1,0}(\mathbf{X}) \cap \mathbb{D}^2_{1,0}(\mathbf{Y})$ as follows.*

$$\mathbf{X} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \quad \mathbf{Y} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} 1 & 1 \\ 0 & 0 \\ 0 & 1 \end{pmatrix},$$

where $\mathbf{D}$ is obtained by deleting the second column from $\mathbf{X}$ and $\mathbf{Y}$. Since more than one row of $\mathbf{X}$ and $\mathbf{Y}$ are different, we see that $\mathbb{D}^2_{0,1}(\mathbf{X}) \cap \mathbb{D}^2_{0,1}(\mathbf{Y}) = \emptyset$ and therefore the equivalence does not hold.

Given this result, we show that the insertion/deletion equivalence holds if one fixes a number of insdel for each dimension to be deleted.

Lemma 3. *For positive integers $m_1, \ldots, m_d$, $i \in [d]$, a vector $\boldsymbol{t} = (t_1, \ldots, t_d) \in \mathbb{N}^d$, and any two arrays $\mathbf{X}, \mathbf{Y} \in \Sigma_q^{\bigotimes_{\ell=1}^d m_\ell}$ it holds that,*

$$\mathbb{D}^d_{\boldsymbol{t}}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{t}}(\mathbf{Y}) \neq \emptyset \Leftrightarrow \mathbb{ID}^d_{\boldsymbol{t}}(\mathbf{X}) \cap \mathbb{ID}^d_{\boldsymbol{t}}(\mathbf{Y}) \neq \emptyset.$$

Proof. We only show the "only if" part, since the "if" part follows by similar arguments. Let $\boldsymbol{t}^{\text{ins}} = (t_1^{\text{ins}}, t_2^{\text{ins}}, \ldots, t_d^{\text{ins}})$ and $\boldsymbol{t}^{\text{del}} = (t_1^{\text{del}}, t_2^{\text{del}}, \ldots, t_d^{\text{del}})$ such that $\boldsymbol{t} = \boldsymbol{t}^{\text{ins}} + \boldsymbol{t}^{\text{del}}$. Assume that there exists an array $\mathbf{I} \in \Sigma_q^{\bigotimes_{i=1}^d (m_i + (t_i^{\text{ins}} - t_i^{\text{del}}))}$ such that $\mathbf{I} \in \mathbb{ID}^d_{\boldsymbol{t}}(\mathbf{X}) \cap \mathbb{ID}^d_{\boldsymbol{t}}(\mathbf{Y})$. The order of deletions and insertions matters here, therefore we define $\mathbf{X}'$ and $\mathbf{Y}'$ to be the arrays resulting from $\boldsymbol{t}^{\text{del}}$-deletion, i.e., it holds that $\mathbf{X}' \in \mathbb{D}^d_{\boldsymbol{t}^{\text{del}}}(\mathbf{X})$ and $\mathbf{Y}' \in \mathbb{D}^d_{\boldsymbol{t}^{\text{del}}}(\mathbf{Y})$. It then follows that $\mathbf{I} \in \mathbb{I}^d_{\boldsymbol{t}^{\text{ins}}}(\mathbf{X}') \cap \mathbb{I}^d_{\boldsymbol{t}^{\text{ins}}}(\mathbf{Y}')$. By Theorem 4, there exists an array $\mathbf{D} \in \Sigma_q^{\bigotimes_{i=1}^d (m_i - (t_i^{\text{ins}} + t_i^{\text{del}}))}$ such that $\mathbf{D} \in \mathbb{D}^d_{\boldsymbol{t}^{\text{ins}}}(\mathbf{X}') \cap \mathbb{D}^d_{\boldsymbol{t}^{\text{ins}}}(\mathbf{Y}')$ and as a result $\mathbf{D} \in \mathbb{D}^d_{\boldsymbol{t}}(\mathbf{X}) \cap \mathbb{D}^d_{\boldsymbol{t}}(\mathbf{Y})$. $\square$

References

[1] R. Heckel, G. Mikutis, and R. N. Grass, "A characterization of the dna data storage channel," *Scientific reports*, vol. 9, no. 1, pp. 1–12, 2019.

[2] T. Buschmann and L. V. Bystrykh, "Levenshtein error-correcting barcodes for multiplexed dna sequencing," *BMC Bioinformatics*, vol. 14, no. 1, pp. 1–10, 2013.

[3] A. S. J. Helberg, *Coding for the correction of synchronization errors*. PhD thesis, Randse Afrikaanse Universiteit, 1993.

[4] F. Sala, C. Schoeny, N. Bitouzé, and L. Dolecek, "Synchronizing files from a large number of insertions and deletions," *IEEE Transactions on Communications*, vol. 64, no. 6, pp. 2258–2273, 2016.

[5] Y. M. Chee, H. M. Kiah, A. Vardy, and E. Yaakobi, "Coding for racetrack memories," *IEEE Transactions on Information Theory*, vol. 64, no. 11, pp. 7094–7112, 2018.

[6] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," in *Soviet Physics Doklady*, vol. 10, pp. 707–710, 1966.

[7] D. Cullina and N. Kiyavash, "An improvement to levenshtein's upper bound on the cardinality of deletion correcting codes," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 3862–3870, 2014.

[8] R. R. Varshamov and G. M. Tenengolts, "Codes which correct single asymmetric errors (in Russian)," *Automatika i Telemkhanika*, vol. 161, no. 3, pp. 288–292, 1965.

[9] G. M. Tenengolts, "Nonbinary codes, correcting single deletion or insertion (corresp.)," *IEEE Transactions on Information Theory*, vol. 30, no. 5, pp. 766–769, 1984.

[10] V. Guruswami and C. Wang, "Deletion codes in the high-noise and high-rate regimes," *IEEE Transactions on Information Theory*, vol. 63, pp. 1961–1970, Apr. 2017.

[11] J. Brakensiek, V. Guruswami, and S. Zbarsky, "Efficient low-redundancy codes for correcting multiple deletions," *IEEE Transactions on Information Theory*, vol. 64, no. 5, pp. 3403–3410, 2017.

[12] S. K. Hanna and S. El Rouayheb, "Guess & check codes for deletions, insertions, and synchronization," *IEEE Transactions on Information Theory*, vol. 65, no. 1, pp. 3–15, 2018.

[13] R. Gabrys and F. Sala, "Codes correcting two deletions," *IEEE Transactions on Information Theory*, vol. 65, pp. 965–974, Feb 2019.

[14] J. Sima, N. Raviv, and J. Bruck, "Two deletion correcting codes from indicator vectors," *IEEE Transactions on Information Theory*, vol. 66, no. 4, pp. 2375–2391, 2020.

[15] J. Sima and J. Bruck, "On optimal k-deletion correcting codes," *IEEE Transactions on Information Theory*, vol. 67, no. 6, pp. 3360–3375, 2021.

[16] V. Guruswami and J. Håstad, "Explicit two-deletion codes with redundancy matching the existential bound," in *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 21–32, SIAM, 2021.

[17] J. Sima, R. Gabrys, and J. Bruck, "Optimal systematic t-deletion correcting codes," *IEEE International Symposium on Information Theory (ISIT)*, 2020.

[18] A. Krishnamurthy, A. Mazumdar, A. McGregor, and S. Pal, "Trace reconstruction: Generalized and parametrized," *European Symposium on Algorithms*, September 2019.

[19] S. Bakirtas and E. Erkip, "Database matching under column deletions," *arXiv preprint arXiv:2105.09616*, 2021.

[20] R. Bitar, L. Welter, I. Smagloy, A. Wachter-Zeh, and E. Yaakobi, "Criss-cross insertion and deletion correcting codes," *IEEE Transactions on Information Theory*, vol. 67, no. 12, pp. 7999–8015, 2021.

[21] Y. M. Chee, M. Hagiwara, and V. Van Khu, "Two dimensional deletion correcting codes and their applications," *IEEE International Symposium on Information Theory (ISIT)*, 2021.

[22] M. Hagiwara, "Conversion method from erasure codes to multi-deletion error-correcting codes for information in array design," *International Symposium on Information Theory and Its Applications (ISITA)*, 2020.

[23] L. Welter, R. Bitar, A. Wachter-Zeh, and E. Yaakobi, "Multiple criss-cross insertion and deletion correcting codes," *IEEE Transactions on Information Theory (Early Access)*, 2022.