# Exact Reconstruction from Insertions in Synchronization Codes

Frederic Sala, Ryan Gabrys, Clayton Schoeny, and Lara Dolecek



**Abstract**

This work studies problems in data reconstruction, an important area with numerous applications. In particular, we examine the reconstruction of binary and nonbinary sequences from synchronization (insertion/deletion-correcting) codes. These sequences have been corrupted by a fixed number of symbol insertions (larger than the minimum edit distance of the code), yielding a number of distinct traces to be used for reconstruction. We wish to know the minimum number of traces needed for exact reconstruction. This is a general version of a problem tackled by Levenshtein for uncoded sequences.

We introduce an exact formula for the maximum number of common supersequences shared by sequences at a certain edit distance, yielding an upper bound on the number of distinct traces necessary to guarantee exact reconstruction. Without specific knowledge of the codewords, this upper bound is tight. We apply our results to the famous single deletion/insertion-correcting Varshamov-Tenengolts (VT) codes and show that a significant number of VT codeword pairs achieve the worst-case number of outputs needed for exact reconstruction. This result opens up a novel area for study: the development of codes with comparable rate and minimum distance properties that require fewer traces for reconstruction.

**Index Terms**

Insertions and deletions; sequence reconstruction; Levenshtein distance; synchronization codes.

## I. Introduction

This paper is concerned with the problem of reconstructing sequences (selected from error-correcting codes) from traces. Our main result is that exact reconstruction is always possible given $M$ traces (formed by a $t$-insertion channel) of a length-$n$ $q$-ary codeword in an $\ell$-insertion/deletion-correcting code for

$$M \geqslant \sum_{j=\ell+1}^{t} \sum_{i=0}^{t-j} \binom{2j}{j} \binom{t+j-i}{2j} \binom{n+t}{i} (q-1)^i (-1)^{t+j-i} + 1. \tag{1}$$

Without further knowledge of the code, the bound in (1) is tight. In other words, if $M$ is smaller than the right hand side of (1), there exists a pair of sequences of length $n$ with $M$ common traces, so that reconstruction cannot be guaranteed, and, moreover, these sequences are at distance at least $2\ell+1$, and could thus be part of an $\ell$-insertion/deletion-correcting code. The result generalizes the uncoded version from [27] and [26], which can be recovered by taking $\ell = 1$ in (1) and simplifying the result through a series of combinatorial identities. It is surprising that exact formulas like (1) exist, given the paucity of exact expressions in insertion and deletion problems. Before we further discuss results such as (1), we briefly introduce the context for this work.

Data reconstruction from traces is an important problem with numerous applications, including data recovery, genomics and other areas of biology, chemistry, sensor networks, and many others. The general problem of reconstruction is broadly divided into probabilistic and adversarial variants. In the probabilistic version, the traces are formed by passing the data through a noisy channel (typically an edit channel with certain deletion and insertion probabilities) and the goal is to reconstruct the data to within a certain error probability. In [1], an algorithm is introduced based on bitwise majority alignment that reconstructs an original sequence of length $n$ with high probability from $O(\log n)$ traces when the deletion probability is $O(\log \frac{1}{n})$. These results were extended for the deletion/insertion channel in [3] and improved upon in [4]. Sequence reconstruction with constant deletion probability was also studied in [2], where the authors showed that when the sequence length is $n$, reconstruction is possible, with high probability, from a number of traces polynomial in $n$ in time polynomial in $n$.

The adversarial variant, which we are concerned with in this paper, allows for traces formed from a worst-case number of errors and seeks to determine what is the smallest number of traces needed for zero-error reconstruction [27]. This setup for sequence reconstruction has also been applied to associative memories [22]. In these memories, each entry is associated with neighboring entries; when searching for a particular entry, "clues" are given in the form of such neighboring entries. This notion leads to a generalization of sequence reconstruction; here, the question becomes how many sequences are associated with (i.e., of maximum Hamming distance from) three or more sequences. The resulting intersection is called an *output set*,

F. Sala, C. Schoeny, and L. Dolecek are with the Electrical Engineering Department, University of California, Los Angeles, Los Angeles, CA 90095, USA. (e-mail: {fredsala, cschoeny}@ucla.edu, dolecek@ee.ucla.edu).

R. Gabrys is with Spawar Systems Center Pacific Code 532, San Diego, CA, 92152 (e-mail: ryan.gabrys@navy.mil).

Research supported in part by the NSF Graduate Research Fellowship Program and NSF grant CCF-1527130. Part of the results in this paper were presented at the IEEE International Symposium on Information Theory (ISIT) in 2015 and 2016 (references [30], [31]).

and the size of the maximum output set is the *uncertainty* of the memory. This line of research was extended by in [23], which studies efficient codes for information retrieval in memories with small uncertainty, and in [24], where the number of input clues is varied. We note that all of these works target the Hamming metric.

By contrast, we are specifically interested in the following problem: if a codeword from a synchronization (specifically, insertion/deletion-correcting) code, i.e., a code with a certain minimum edit/Levenshtein distance, is repeatedly transmitted through a noisy channel, how many distinct channel outputs (traces) are necessary for zero-error reconstruction? This question is indeed meaningful; consider, for example, phylogenomics, where we wish to reconstruct the genetic sequence of an ancestor organism from a large number of sequences of evolutionary descendants. Each of the descendant sequences is formed from a number of base pair insertions. The related question of how to efficiently perform the reconstruction is tackled, from a coding-theoretic point of view in the recent work [6].

Moreover, reconstruction of encoded data has a natural application to data storage. We can partition the operational lifetime of a disk drive, memory, or other data storage device into two periods. The first period is regular, short-term use; popular devices and common error-correcting codes all target this scenario. Here, a small number (often one) of channel outputs are used to read the data. The second period refers to extremely long-term use of the device, well beyond the guaranteed operating lifetime. In this case, many reads can be performed, resulting in a large number of traces that can be used to recover the original data. This type of very long term use is increasingly relevant. For example, DNA storage is targeted as a medium to store data for $10^4$ or more years, and, indeed, over the long term, DNA sequences are affected by insertions and other errors that can be modeled by insertions (duplications, tandem/block duplications, block insertions) [5]. The present paper studying reconstruction from insertions can therefore be viewed as complementary to the many recent efforts studying coding for data storage in DNA [7]–[15]. Of course, our work also joins recent coding-theoretic studies on insertions and deletions, such as [18]–[20].

In a classic paper, [27], Levenshtein explored several variations of the reconstruction problem, studying both adversarial and probabilistic channels and exact and approximate reconstruction. However, the problem of reconstructing sequences affected by insertions and deletions in the case where the sequences are themselves part of a code (e.g., have a certain minimum edit distance) was left open. We tackle this problem for the insertion case in the current work. We target insertions for two reasons. First, insertions are edit errors, which are of particular interest as we often wish to reconstruct strings. In keeping with our biology theme, as described above, we note that insertions are a common type of mutation affecting genetic sequences. Second, unlike deletions, insertions offer symmetries that allow a tractable search for exact formulas.

The remainder of this paper is organized in the following way. In the next section, we introduce our problem setup, discuss prior work, and describe notation. In Section III, we prove a result on the common supersequences problem for the binary case. In Section IV, we prove and interpret the more general, non-binary result. We also discuss important special cases and corollaries. In Section V, we apply our result to the single deletion/insertion-correcting Varshamov-Tenengolts (VT) codes. Finally, in Section VI, we consider extensions to the deletion and insertion/deletion channels. We conclude the paper in Section VII.

## II. Preliminaries

### A. Problem setup

Levenshtein observed in [27] that given a sequence $X \in V \subseteq \mathbb{F}_q^n$ for a set $V$ and a finite field $\mathbb{F}_q$, it is always possible to exactly reconstruct $X$ given $N_q^H(V,t)+1$ distinct elements of $B_t(X,H)$, the ball produced by applying up to $t$ single errors from a set of error functions $H$ to $X$. Here, $N_q^H(V,t)$ is defined by

$$N_q^H(V,t) = \max_{X,Z \in V, X \neq Z} |B_t(X,H) \cap B_t(Z,H)|.$$

In other words, the problem of exact reconstruction of sequences can be identified with the combinatorial problem of counting (the maximum number of) common distorted sequences. In [27], expressions were given for $N_q^H(V,t)$ for many choices of error sets $H$. In this paper, we focus specifically on the case of insertion channels, so that $H$ is made up of single symbol insertions, and we wish to reconstruct $X$ from its supersequences (sequences formed from $X$ by insertions.) In [27], an expression was provided for $N_q^H(V,t)$ in this scenario, but only for the specific case of $V = \mathbb{F}_q^n$, the **uncoded** case. For this problem setup, we may write the balls $B_t(X,H)$ as insertion balls $I_t(X)$ and denote the expression $N_q^H(\mathbb{F}_q^n,t)$ as $N_q^+(n,t)$. In [27], $N_q^+(n,t)$ was shown to be

$$N_q^+(n,t) = \sum_{i=0}^{t-1} \binom{n+t}{i} (q-1)^i (1-(-1)^{t-i}). \tag{2}$$

However, the problem of reconstruction given a code $V$ differing from the entire set $\mathbb{F}_q^n$ was left open. We tackle this problem in the present work. Consider, for example, reconstructing a sequence that is a member of an $(\ell-1)$-insertion-correcting code $\mathcal{C}'$. Sequences that are part of such codes must have a minimum edit (Levenshtein) distance of $2\ell$ (we make this terminology precise later on.) If this is the case, we can always perform exact reconstruction if we know $N_q^+(\mathcal{C}',t)+1$ distinct supersequences of $X$, where

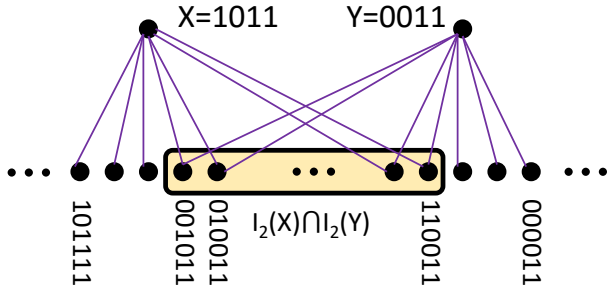


Fig. 1. Example setup for our problem. Here, we are counting common supersequences of $X = 1011$ and $Y = 0011$ produced by $t = 2$ bit insertions. We have that $|I_2(X) \cap I_2(Y)| = 12$.

$$N_q^+(\mathcal{C}', t) = \max_{X,Z \in \mathcal{C}', X \neq Z} |I_t(X) \cap I_t(Z)|.$$

Computing this maximal intersection requires knowing the structure of the code $\mathcal{C}'$. This is challenging, since few such codes are known outside of the single insertion case. Instead, we focus on deriving an expression for the maximum number of common supersequences for sequences at a minimum particular Levenshtein (edit) distance $2\ell$,

$$N_q^+(n, t, \ell) = \max_{\substack{X,Z \in \mathbb{F}_q^n \\ d_L(X,Z) \geqslant 2\ell}} |I_t(X) \cap I_t(Z)|.$$

An example is shown in Figure 1 for $n = 4$ and $t = 2$. Specifically, we prove that $N_q^+(n, t, \ell)$ is given by

$$\sum_{j=\ell}^{t} \sum_{i=0}^{t-j} \binom{2j}{j} \binom{t+j-i}{2j} \binom{n+t}{i} (q-1)^i (-1)^{t+j-i}.$$

Evaluating the above expression provides an upper bound on the number of channel outputs needed for reconstruction of codewords in insertion-correcting codes. Without specifying a particular code construction, this bound is the best possible. Note that Levenshtein's formula $N_q^+(n, t)$ in (2) can be written as

$$N_q^+(n, t) = \max_{\ell \geqslant 1} N_q^+(n, t, \ell).$$

We will see, in fact, that this maximum is always attained at $\ell = 1$. In other words, the maximum number of common supersequences occurs for sequences that are as "close" as possible.

As part of our study, we provide an even more general version of this result where we allow the sequences $X$ and $Z$ to have different lengths. This result can be interpreted as a double generalization of Levenshtein's formula $N_q^+(n, t)$.

### *B. Notation*

We introduce some useful notation. Let $\mathbb{F}_q$ denote the set $\{0, 1, \ldots, q-1\}$ for $q \geqslant 2$ and $[a, b]$ denote the set $\{a, a+1, a+2, \ldots, b-1, b\}$ if $a \leqslant b$. If $q = 2$, we use $\bar{a}$ to denote the complementary symbol in $\mathbb{F}_2$, so that $\bar{a} = 0$ if $a = 1$ and $\bar{a} = 1$ if $a = 0$. We denote sequences with upper-case letters and individual symbols with lower-case letters, so that, for example, $X = x_1 x_2 \ldots x_n \in \mathbb{F}_q^n$ while $x_i \in F_q$ for $1 \leqslant i \leqslant n$. We write $XY$ for the concatenation of sequences $X$ and $Y$; similarly, $aX$ is the concatenation of a symbol $a$ with sequence $X$. We sometimes use the notation $XS$ where $S$ is a set. In this case, $XS$ refers to the set that contains the concatenation of $X$ with all sequences in $S$, $\{XY : Y \in S\}$.

We use the generalized binomial coefficient: for $a, b \in \mathbb{Z}$, $\binom{a}{b} = a(a-1)\ldots(a-b+1)/b!$. We assume $0! = 1$. We set $\binom{a}{b} = 0$ for $b < 0$. We also use the convention $\binom{a}{0} = 1$ for all $a \in \mathbb{Z}$ and $\binom{a}{b} = 0$ if $a = 0$ and $b > 0$. We sometimes rely on the useful fact that $\binom{a}{b} = 0$ if $a > 0$ and $a < b$.

If $n \geqslant v$, $Z \in \mathbb{F}_q^{n-v}$ is a **subsequence** of $X \in \mathbb{F}_q^n$ if $Z$ can be formed from $X$ by deleting $v$ symbols. If $n = v$, $Z$ is the empty sequence, with length 0. Similarly, for $t \geqslant 0$, $W \in \mathbb{F}_q^{n+t}$ is a **supersequence** of $X$ if it can be formed by $t$ symbol insertions into $X$. The set of all length $n - v$ subsequences of $X$ (also called the $v$-deletion ball centered at $X$) is denoted $D_v(X)$; similarly, the set of all length $n + t$ supersequences of $X$ (the $t$-insertion ball centered at $X$) is written $I_t(X)$.

In general, the size of $D_v(X)$ depends on the sequence $X$. For example, $|D_1(X)| = \tau(X)$, where $\tau(X)$ is the number of maximal-length runs of identical symbols in $X$. On the other hand, $|I_t(X)|$ does not depend on $X$ for any $t \geqslant 0$. There is a formula for the size of the supersequence set that only depends on $n, t$ and the alphabet size $q$ [25],

$$|I_t(X)| := I_q(n,t) = \sum_{i=0}^{t} \binom{n+t}{i} (q-1)^i. \tag{3}$$

The distance between sequences $X, Y$ can be measured with the **Levenshtein distance** (or edit distance) $d_L(X,Y)$. This distance is defined in the following way: $d_L(X,Y) = k$ if $k$ is the smallest number of insertions and deletions that can be used to transform $X$ to $Y$. Note that it is not necessary that $X$ and $Y$ have the same length for $d_L(X,Y)$ to be defined. For example, take $X = 00$ and $Y = 010$. Then, $d_L(X,Y) = 1$, since we require just one insertion of a 1 into $X$ to form $Y$. If $X = 110$ and $Y = 000$, then $d_L(X,Y) = 4$. Note that our definition of edit distance does not include substitutions.

An $t$-insertion-correcting code $\mathcal{C}$ is a subset of $\mathbb{F}_q^n$ such that if $X, Y \in \mathcal{C}$ and $X \neq Y$, then $I_t(X) \cap I_t(Y) = \emptyset$. This is equivalent to requiring that $\mathcal{C}$ has minimum Levenshtein distance[1] $2t+2$. We also note that a $t$-insertion-correcting code is also a $t$-deletion-correcting code (and also a $a$-deletion/$b$-insertion-correcting code for any pair $(a,b)$ with $a + b \leqslant t$.) This equivalence was proved in [25].

As described, we are concerned with computing the maximum number of common supersequences between sequences with Levenshtein distance of at least $2\ell$ for $\ell \geqslant 1$. That is, we are interested in the quantity[2] $N_q^+(n,t,\ell)$ defined as

$$N_q^+(n,t,\ell) = \max_{\substack{X,Y \in \mathbb{F}_q^n \\ d_L(X,Y) \geqslant 2\ell}} |I_t(X) \cap I_t(Y)|.$$

We refer to $n$, $t$, and $\ell$ as the *length*, *insertion*, and *distance* arguments, respectively.

Additionally, in our results, we consider a more general version of the problem where the sequences need not be of the same length. One of the two sequences ($Y$) continues to be of length $n$ while the common supersequences remain of length $n+t$. However, we allow $X$ to be of length $n+t-k$ (that is, longer than $Y$ by $t-k$ symbols.) As a result, we make only $k$ insertions into $X$. (Note that we now have two insertion arguments, $t$ and $k$.) Similarly, the distance between $X$ and $Y$ is now required to be $t-k+2\ell$ in order to make up for the additional distance between the sequences resulting from the differing lengths. Observe that $t \geqslant k \geqslant \ell$ in this setup. The goal, then, is to compute

$$N_q^+(n,t,k,\ell) = \max_{\substack{X \in \mathbb{F}_q^{n+t-k}, Y \in \mathbb{F}_q^n \\ d_L(X,Y) \geqslant t-k+2\ell}} |I_k(X) \cap I_t(Y)|.$$

We can always retrieve $N_q^+(n,t,\ell)$ from $N_q^+(n,t,k,\ell)$ by taking $t = k$.

### C. Basic Claims

We use several simple claims as building blocks in our work. First, the following fact is an immediate consequence of our definitions.

**Claim 1.** *For $\ell \geqslant 1$ and $n, t, k$ non-negative integers with $t \geqslant k \geqslant \ell$,*

$$N_q^+(n,t,k,\ell) \leqslant N_q^+(n,t,k,\ell-1).$$

*Proof:* Any two sequences $X, Y$ with distance at least $t-k+2\ell$ also have distance at least $t-k+2(\ell-1)$, so therefore the maximum number of common supersequences for distance argument $\ell-1$ is at least that for distance argument $\ell$. ∎

We also have another easy fact regarding distances.

**Claim 2.** *Let $X' \in \mathbb{F}_q^m$ and $Y' \in \mathbb{F}_q^n$ with $m, n > 0$. If $d_L(X', Y') = v$ and $X' = x_1 X$, then*

$$d_L(X, Y') \in \{v-1, v+1\}.$$

*Proof:* Clearly, $d_L(X,Y')$ cannot be smaller than $v-1$, or we could form $X$ from $Y'$ with fewer than $v-1$ operations and insert $x_1$, retrieving $X'$ in fewer than $v$ operations, a contradiction. Similarly, $d_L(X,Y')$ cannot be larger than $v+1$, since we can form $X'$ from $Y'$ and delete $x_1$ in $v+1$ operations. Lastly, since $X'$ and $X$ differ in length by 1, $d_L(X',Y')$ and $d_L(X,Y')$ cannot have the same parity. ∎

[1]The required minimum distance is $2t+1$; however, since all the codewords in $\mathcal{C}$ are of the same length, the distance between any codeword pair must be even, since to go from one codeword to another, there must be a deletion for every insertion. Thus the minimum distance is in fact $2t+2$. For example, the minimum Levenshtein distance of the single deletion/insertion-correcting Varshamov-Tenegolts codes is 4.

[2]The "+" symbol denotes the fact that we are performing insertions.

Next, we introduce two useful results. First, we have an observation that Levenshtein originally made in [26]. Consider some sequence $Z = z_1 z_2 \dots z_n$. Then, $I_t(Z)$ is the union of two disjoint sets: the set of sequences starting with $z_1$ (which can be formed by placing all $t$ insertions into $z_2 \dots z_n$) and the set of sequences starting with the element $x \in \mathbb{F}_q \setminus z_1$ (which require $x$ to be inserted at the head of $Z$, leaving $t-1$ remaining insertions into $Z$.) Formally, we have that

**Claim 3.** *If $Z = z_1 z_2 \dots z_n \in \mathbb{F}_q^n$ is a sequence and $t \geqslant 1$, then,*

$$I_t(Z) = z_1 I_t(z_2 z_3 \dots z_n) \bigcup \cup_{x \in \mathbb{F}_q \setminus z_1} x I_{t-1}(Z). \tag{4}$$

*If $q = 2$,*

$$I_t(Z) = z_1 I_t(z_2 z_3 \dots z_n) \bigcup \bar{z_1} I_{t-1}(Z). \tag{5}$$

Here, recall that $xI_{t-1}(Z)$ refers to appending all the sequences in $I_{t-1}(Z)$ to the element $x$. The idea in Claim 3 can be exploited to derive recursive expressions for the number of common supersequences. A variant of the following observation was used by Levenshtein in [26]; we provide a proof for completeness.

**Claim 4.** *Let $n$ be a positive integer, $t, k$ be non-negative integers with $t \geqslant k$, and $X' \in \mathbb{F}_q^{n+t-k}, Y' \in \mathbb{F}_q^n$. Write $X' = aX$ and $Y' = bY$ with $a, b \in \mathbb{F}_q$. Then, if $a = b$,*

$$|I_k(X') \cap I_t(Y')| = |I_k(X) \cap I_t(Y)| + (q-1)|I_{k-1}(aX) \cap I_{t-1}(aY)|. \tag{6}$$

*If $a \neq b$,*

$$|I_k(X') \cap I_t(Y')| = |I_k(X) \cap I_{t-1}(bY)| + |I_{k-1}(aX) \cap I_t(Y)| + (q-2)|I_{k-1}(aX) \cap I_{t-1}(bY)|. \tag{7}$$

*If $q = 2$, the formulas (6) and (7) reduce to*

$$|I_k(X') \cap I_t(Y')| = |I_k(X) \cap I_t(Y)| + |I_{k-1}(aX) \cap I_{t-1}(aY)|, \tag{8}$$

*and*

$$|I_k(X') \cap I_t(Y')| = |I_k(X) \cap I_{t-1}(\bar{a}Y)| + |I_{k-1}(aX) \cap I_t(Y)|, \tag{9}$$

*respectively.*

*Proof:* First we consider the case of $a = b$. A common supersequence $W'$ of $X' = aX$ and $Y' = aY$ either starts with $a$ or an element in $\mathbb{F}_q \setminus \{a\}$. If $W'$ starts with $a$, we write $W' = aW$. Using Claim 3, $W$ can be formed by $k$ insertions into $X$ and $t$ insertions into $Y$, so $W$ is a common supersequence of $X$ and $Y$. That is, it is in the set $I_k(X) \cap I_t(Y)$. Similarly, if $W' = w_1 W$ starts with $w_1$, one of the $q-1$ elements in $\mathbb{F}_q \setminus \{a\}$, it must be formed by inserting $w_1$ at the head of $X' = aX$ and at the head of $Y' = aY$. Therefore, $W$ is in $I_{t-1}(aX) \cap I_{t-1}(bY)$. There are thus $(q-1) \times |I_{k-1}(aX) \cap I_{t-1}(aY)|$ choices for such supersequences $W'$. This establishes (6).

For the case of $a \neq b$, if $W' = aW$, $W$ can be formed from $X'$ by inserting $k$ elements into $X$, while $W'$ can be formed from $Y'$ by inserting $a$ at the head and $t-1$ more elements into $Y' = bY$. If $W' = bW$, it is formed from $X'$ by inserting $b$ at the head and $k-1$ elements into $X' = aX$ while $W$ is formed from $Y'$ by inserting $t$ elements into $Y$. Lastly, if $W'$ starts with $w_1$, one of the $q-2$ elements in $\mathbb{F}_q \setminus \{a, b\}$, it is formed from $X'$ by inserting $w_1$ at the head and $k-1$ more elements into $X' = aX$ and from $Y'$ by inserting $w_1$ at the head and $t-1$ more elements into $Y' = bY$. The sets given by the three possibilities are disjoint, giving (7). ∎

## III. Maximum number of common supersequences: binary case

The purpose of this section is to introduce a formula for $N_2^+(n, t, k, \ell)$ and to provide implications and a proof. We introduce the binary result first as a gentle introduction and study the more general case for $q > 2$ in the following section.

**Theorem 5.** *Let $n$ be a positive integer and $t, k, \ell$ be non-negative integers such that $t \geqslant k \geqslant \ell$ and $n \geqslant \ell$. Then, we have the formula*

$$N_2^+(n, t, k, \ell) = \sum_{j=\ell}^{k} \binom{t-k+2j}{j} \binom{n+k-(2j+1)}{k-j}, \tag{10}$$

*and, in the equal length sequence case $t = k$, we have that*

$$N_2^+(n, t, \ell) = \sum_{j=\ell}^{t} \binom{2j}{j} \binom{n+t-(2j+1)}{t-j}. \tag{11}$$

We begin with some observations on Theorem 5. We are more interested in the formula in (11) compared to that in (10) because most insertion/deletion codes have equal-length codewords. We will later see (from the proof of Theorem 5) that the

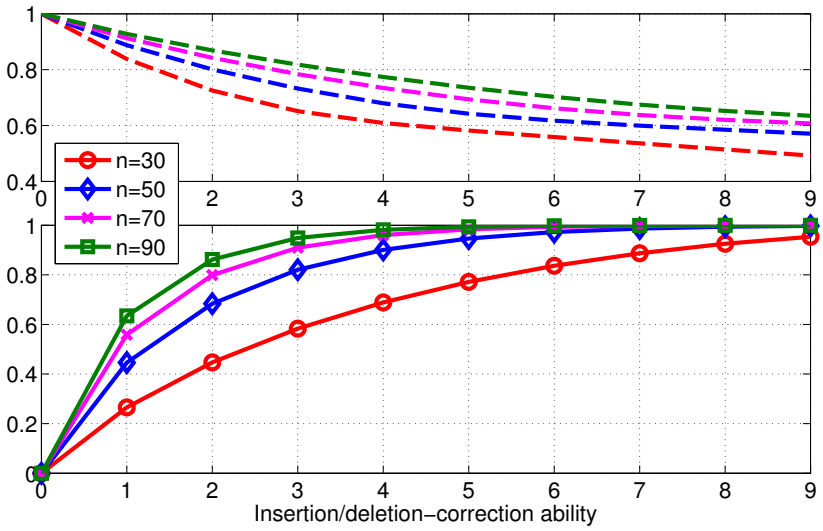


Fig. 2. Curves showing insertion/deletion code rates (dashed lines) and reconstruction requirement reduction percentage (solid lines) given traces affected by $t = 15$ insertions for codes of lengths $n = 30, 50, 70, 90$ capable of correcting of $0, 1, \ldots, 9$ edit (insertion or deletion) errors.

sequences that yield the maximum $N_2^+(n,t,\ell)$ common supersequences are those at distance precisely $2\ell$. This confirms the intuitive idea that the maximum number of common supersequences is monotonically decreasing in the distance argument $\ell$.

Results in the spirit of (11) encourage us to examine the *tradeoff between code rate and reconstruction requirements*. For example, the expression in (11) is decreasing in $\ell$, the code's insertion/deletion-correcting ability. (Note that some of the terms $\binom{n+t-(2j+1)}{t-j}$ can be negative for sufficiently large $t$ and $j$, but we can only increase $\ell$ up to $n$, and in this regime, all such terms are positive.) Increasing $\ell$ allows us to reconstruct with fewer and fewer traces, but this comes at the cost of decreased code rate. We show an example of this tradeoff for insertion/deletion correcting codes of lengths $n = 30, 50, 70, 90$ in Figure 2. Here, we plot two curves for each code based on the error-correcting ability; the dashed curves show code rates (based on non-asymptotic upper bounds from [19]), while the thick curves show the percentage reduction (normalized to 1) in the number of traces needed to guarantee exact reconstruction given traces formed by $t = 15$ random symbol insertions.

Another consequence of our results is that we can recover Levenshtein's formula $N_2^+(n,t)$ in [26] by taking $\ell = 1$ in (11) and applying the binomial identity

$$\sum_{j=1}^{t} \binom{2j}{j} \binom{n+t-2j-1}{t-j} = 2 \sum_{i=0}^{t-1} \binom{n+t}{i}.$$

Next we provide a roadmap for the proof of Theorem 5. First, we define

$$\mathcal{N}_2^+(n,t,k,\ell) := \sum_{j=\ell}^{k} \binom{t-k+2j}{j} \binom{n+k-(2j+1)}{k-j}.$$

Then, our goal becomes to prove that $N_2^+(n,t,k,\ell) = \mathcal{N}_2^+(n,t,k,\ell)$. We show that the formula given by $\mathcal{N}_2^+(n,t,k,\ell)$ satisfies two recursions: first, $\mathcal{N}_2^+(n,t,k,\ell) = \mathcal{N}_2^+(n-1,t,k,\ell) + \mathcal{N}_2^+(n,t-1,k-1,\ell)$, and second, $\mathcal{N}_2^+(n,t,k,\ell) = \mathcal{N}_2^+(n,t-1,k,\ell) + \mathcal{N}_2^+(n-1,t,k-1,\ell-1)$. This will be done purely through combinatorial manipulations of the formula given by $\mathcal{N}_2^+(n,t,k,\ell)$. Afterwards, we show that the maximization given by $N_2^+(n,t,k,\ell)$ satisfies two nearly identical inequalities depending on whether the maximizing sequences start with the same first bit or a differing first bit: $N_2^+(n,t,k,\ell) \leqslant N_2^+(n-1,t,k,\ell) + N_2^+(n,t-1,k-1,\ell)$ and $N_2^+(n,t,k,\ell) \leqslant N_2^+(n,t-1,k,\ell) + N_2^+(n-1,t,k-1,\ell-1)$, respectively. We do this by exploiting Claim 4. These two results are applied in an inductive argument to show that $N_2^+(n,t,k,\ell) \leqslant \mathcal{N}_2^+(n,t,k,\ell)$. All that remains is to give a pair of sequences that yield equality in this formula. We will show that

$$X = \underbrace{00\ldots0}_{t-k+n\ 0\text{'s}} \quad \text{and} \quad Y = \underbrace{11\ldots1}_{\ell\ 1\text{'s}} \underbrace{00\ldots0}_{n-\ell\ 0\text{'s}}.$$

are two such sequences.

We also briefly discuss two important improvements of our proof technique compared to that of Levenshtein for the $N_q^+(n,t)$ result. First, we generalize the problem to the different-lengths case where $t$ need not be equal to $k$. This enables us to involve the second recursion (for sequences starting with different bits) directly in the induction. This was not possible in Levenshtein's proof, as the second recursion immediately breaks down into different-length cases (and formulas for such cases were not known); however, for $\ell = 1$, this issue can be overcome. In the cases $\ell > 1$, this is not possible. Interestingly, our approach

mirrors some proofs in combinatorics, where it is easier to prove a general formula compared to a special case. In addition, we note that unlike in Levenshtein's proof, we require a careful accounting of the recursions' effects on the distance.

The proof of the following lemma revolves around tedious manipulations and is deferred to the appendix.

**Lemma 6.** *For $n$ a positive integer and $t, k, \ell$ non-negative integers with $t \geqslant k \geqslant \ell$,*

$$\mathcal{N}_2^+(n,t,k,\ell) = \mathcal{N}_2^+(n-1,t,k,\ell) + \mathcal{N}_2^+(n,t-1,k-1,\ell), \tag{12}$$

*and*

$$\mathcal{N}_2^+(n,t,k,\ell) = \mathcal{N}_2^+(n,t-1,k,\ell) + \mathcal{N}_2^+(n-1,t,k-1,\ell-1). \tag{13}$$

Next, we show that the maximization $N_2^+(n,t,k,\ell)$ satisfies similar recursions to those of Lemma 6:

**Lemma 7.** *Let $n$ be a positive integer and $t, k, \ell$ be non-negative integers such that $t \geqslant k \geqslant \ell$. Let $X' = aX, Y' = bY$ be two sequences satisfying $X' \neq Y'$[3], $X' \in \mathbb{F}_2^{n+t-k}, Y' \in \mathbb{F}_2^n$, and $d_L(X',Y') = t-k+2\ell$. Then, if $a = b$,*

$$|I_k(X') \cap I_t(Y')| \leqslant N_2^+(n-1,t,k,\ell) + N_2^+(n,t-1,k-1,\ell), \tag{14}$$

*and if $a \neq b$,*

$$|I_k(X') \cap I_t(Y')| \leqslant N_2^+(n,t-1,k,\ell) + N_2^+(n-1,t,k-1,\ell-1). \tag{15}$$

*Proof:* We are given sequences $X', Y'$ satisfying $X' \neq Y'$, $X' \in \mathbb{F}_2^{n+t-k}, Y' \in \mathbb{F}_2^n$, and $d_L(X',Y') = t-k+2\ell$. We have $X' = aX$ and $Y' = bY$, with $a, b \in \mathbb{F}_2$. There are two cases to consider. First, we have the case where $a = b$. From (8),

$$|I_k(aX) \cap I_t(bY)| = |I_k(aX) \cap I_t(aY)| = |I_k(X) \cap I_t(Y)| + |I_{k-1}(aX) \cap I_{t-1}(aY)|. \tag{16}$$

We note that since $d_L(X',Y') = t-k+2\ell$ and $X' = aX$, $Y' = aY$, $X$ and $Y$ must be at the same distance as $X'$ and $Y'$, so that $d_L(X,Y) = t-k+2\ell$. Thus, $X$ is of length $(n-1)+t-k$, $Y$ is of length $n-1$, and $d_L(X,Y) = t-k+2\ell$. Then, we have that $|I_k(X) \cap I_t(Y)| \leqslant N_2^+(n-1,t,k,\ell)$. Similarly, we have that $|I_{k-1}(aX) \cap I_{t-1}(aY)| \leqslant N_2^+(n,t-1,k-1,\ell)$. (We call this step *argument matching*, since we are computing the length, insertion, and distance arguments in order to produce the correct $N^+$ bound.) Putting the two bounds into (16) gives

$$|I_k(aX) \cap I_t(bY)| \leqslant N_2^+(n-1,t,k,\ell) + N_2^+(n,t-1,k-1,\ell).$$

Next we have the case where $a \neq b$. Then, from (9),

$$|I_k(aX) \cap I_t(bY)| = |I_k(X) \cap I_{t-1}(bY)| + |I_{k-1}(aX) \cap I_t(Y)|. \tag{17}$$

Again, we bound the terms in (17) with formulas of the type $N_2^+(n,t,k,\ell)$. In this case, the argument matching is slightly more complicated. For the first term in (17), $bY$ has length $n$ while the insertion arguments are clearly $t-1$ and $k$. It remains to find the distance argument $\ell'$. Recall that $d_L(aX,bY) = t-k+2\ell$. We have, from Claim 2, that $d_L(X,bY) \in \{t-k+2\ell-1, t-k+2\ell+1\}$ and $d_L(aX,Y) \in \{t-k+2\ell-1, t-k+2\ell+1\}$.We must have $d_L(X,bY) = (t-1)-k+2\ell'$, so that $\ell' = \frac{1}{2}(d_L(X,bY)+k-(t-1)) \in \{\ell, \ell+1\}$. Thus, the possible argument tuples for the first term are $\{(n,t-1,k,\ell),(n,t-1,k,\ell+1)\}$. Next, for the second term in (17), $Y$ has length $n$ and the insertion arguments are $t$ and $k-1$. The distance argument $\ell'$ satisfies $d_L(aX,Y) = t-(k-1)+2\ell' \in \{t-k+2\ell-1, t-k+2\ell+1\}$ by Claim 2, so that $\ell' \in \{\ell-1, \ell\}$. Then, the possible argument tuples for this term are $\{(n-1,t,k-1,\ell-1),(n-1,t,k-1,\ell)\}$. Next, applying Claim 1, we have that

$$|I_k(X) \cap I_{t-1}(bY)| \leqslant \max\{N_2^+(n,t-1,k,\ell), N_2^+(n,t-1,k,\ell+1)\} = N_2^+(n,t-1,k,\ell),$$

and

$$|I_{k-1}(aX) \cap I_t(Y)| \leqslant \max\{N_2^+(n-1,t,k-1,\ell-1), N_2^+(n-1,t,k-1,\ell)\} = N_2^+(n-1,t,k-1,\ell-1).$$

Plugging these two expressions into (17) yields

$$|I_k(aX) \cap I_t(bY)| \leqslant N_2^+(n,t-1,k,\ell) + N_2^+(n-1,t,k-1,\ell-1),$$

and the proof is complete. ∎

Now that we have established Lemmas 6 and 7, we are ready for the proof of Theorem 5.

*Proof:* We prove, by strong induction on $n+t+k$, that

$$N_2^+(n,t,k,\ell) \leqslant \mathcal{N}_2^+(n,t,k,\ell). \tag{18}$$

Afterward, we give two sequences which meet the equality case of (18), completing the proof.

[3]Clearly $X' \neq Y'$ if $t > k$, since the two sequences have different length. The condition is only necessary in the equal-length sequences case $t = k$.

The base cases are $n+t+k \in \{1,2\}$. We first deal with $n \in \{1,2\}$ and $t=k=\ell=0$. Observe that for any $X, Y \in \mathbb{F}_2$, $I_0(X) \cap I_0(Y) \subseteq \{X\}$, so $|I_0(X) \cap I_0(Y)| \leqslant 1$. Now, $\mathcal{N}_2^+(n,0,0,0)$ evaluates to $\binom{0}{0}\binom{n-1}{0} = 1$, as desired. The other case is $n=1$, $t=1$, and $k=0$, as we require $t \geqslant k$. Again, $I_0(X) \cap I_t(Y) \subseteq \{X\}$ and thus $|I_0(X) \cap I_t(Y)| \leqslant 1$. $\mathcal{N}_2^+(1,1,0,0)$ evaluates to $\binom{0}{0}\binom{n-1}{0} = 1$ and we are done.

Next, we assume that the claim in (18) holds for all $n+t+k < m$. We will show that it is true for $n+t+k=m$, for $m > 2$. Consider sequences $X' = aX, Y' = bY$ where $X' \in \mathbb{F}_2^{n+t-k}$, $Y' \in \mathbb{F}_2^n$, $d_L(X',Y') = t-k+2\ell$, and $n+t+k=m$. First, if $a=b$, we have that

$$\begin{aligned} |I_k(X') \cap I_t(Y')| &\leqslant N_2^+(n-1,t,k,\ell) + N_2^+(n,t-1,k-1,\ell) \\ &\leqslant \mathcal{N}_2^+(n-1,t,k,\ell) + \mathcal{N}_2^+(n,t-1,k-1,\ell) \\ &= \mathcal{N}_2^+(n,t,k,\ell). \end{aligned}$$

In the first inequality, we used (14) from Lemma 7. In the second inequality, we used the induction hypothesis (as $(n-1)+t+k < m$ and $n+(t-1)+(k-1) < m$). In the final equality, we applied the recursion (12) from Lemma 6.

The remaining case $a \neq b$ is nearly identical; we use the expressions (15) and (13) from Lemmas 7 and 6, respectively. We can again apply the induction hypothesis, since $n+(t-1)+k < m$ and $(n-1)+t+(k-1) < m$. We have that

$$\begin{aligned} |I_k(X') \cap I_t(Y')| &\leqslant N_2^+(n,t-1,k,\ell) + N_2^+(n-1,t,k-1,\ell-1) \\ &\leqslant \mathcal{N}_2^+(n,t-1,k,\ell) + \mathcal{N}_2^+(n-1,t,k-1,\ell-1) \\ &= \mathcal{N}_2^+(n,t,k,\ell). \end{aligned}$$

Thus we conclude that indeed $N_2^+(n,t,k,\ell) \leqslant \mathcal{N}_2^+(n,t,k,\ell)$. All that remains is to demonstrate that there exists at least one pair of sequences $X', Y'$ such that $|I_k(X') \cap I_t(Y')| = \mathcal{N}_2^+(n,t,k,\ell)$. We take

$$X' = \underbrace{00\ldots0}_{t-k+n\ 0\text{'s}} \text{ and } Y' = \underbrace{11\ldots1}_{\ell\ 1\text{'s}}\underbrace{00\ldots0}_{n-\ell\ 0\text{'s}}.$$

It is hard to give a direct proof that $|I_k(X') \cap I_t(Y')| = \mathcal{N}_2^+(n,t,k,\ell)$; instead, we proceed inductively. As we will see, the induction is identical to the previous proof, replacing inequalities with equalities. As before, the induction is on $n+t+k$.

The base cases here are $n \in \{1,2\}$ and $t=k=0$, so that $\ell=0$ as well, along with $n=1, t=1, k=0$, and $\ell=0$. The cases of $t=0$ yield $X'=0$ and $Y'=0$ or $X'=00$ and $Y'=00$. Thus, $|I_0(X') \cap I_0(Y')| = 1 = \mathcal{N}_2^+(n,t,k,\ell)$, as desired. If $n=1, t=1$, we have that $X'=00$ and $Y'=0$. Here too, $|I_0(X') \cap I_1(Y')| = 1 = \mathcal{N}_2^+(n,t,k,\ell)$.

Assume that the induction hypothesis holds for $n+t+k < m$; we show it is true for $n+t+k=m$. If $\ell \geqslant 1$, we apply (9) to write

$$|I_k(X') \cap I_t(Y')| = |I_k(X) \cap I_{t-1}(Y')| + |I_{k-1}(X') \cap I_t(Y)|, \tag{19}$$

where $X = \underbrace{00\ldots0}_{t-k+n-1\ 0\text{'s}}$ and $Y = \underbrace{11\ldots1}_{\ell-1\ 1\text{'s}}\underbrace{00\ldots0}_{n-\ell\ 0\text{'s}}$. Note that to produce $X$ from $Y'$ with the fewest operations, we must remove $\ell$ 1's and insert $t-k+n-1-(n-\ell) = \ell+t-k-1$ 0's. Thus, $d_L(X,Y') = t-k+2\ell-1$. A similar calculation gives $d_L(X',Y) = t-k+2\ell-1$. Now we again match arguments: in the first term in (19), $Y'$ has length $n$, the insertion arguments are $t-1$ and $k$ and $d_L(X,Y') = t-k+2\ell-1$. Thus, the distance argument satisfies $\ell' = \frac{1}{2}(d_L(X,Y')+k-(t-1)) = \frac{1}{2}(t-k+2\ell-1+k-(t-1)) = \ell$. Applying the induction hypothesis, we may thus write $|I_k(X) \cap I_{t-1}(Y')| = \mathcal{N}_2^+(n,t-1,k,\ell)$. Similarly, for the second term in (19), $Y$ has length $n-1$, the insertion arguments are $t$ and $k-1$ while $d_L(X',Y) = t-k+2\ell-1$. The distance argument satisfies $\ell' = \frac{1}{2}(d_L(X',Y)+(k-1)-t) = \frac{1}{2}(t-k+2\ell-1+(k-1)-t) = \ell-1$. Again apply the induction hypothesis to write $|I_{k-1}(X') \cap I_t(Y)| = \mathcal{N}_2^+(n-1,t,k-1,\ell-1)$.

We substitute these equalities in (19) and apply Lemma 6, yielding

$$|I_k(X') \cap I_t(Y')| = \mathcal{N}_2^+(n,t-1,k,\ell) + \mathcal{N}_2^+(n-1,t,k-1,\ell-1) = \mathcal{N}_2^+(n,t,k,\ell).$$

If $\ell=0$, $X'$ and $Y'$ both start with 0 so we apply (8) to write

$$|I_k(X') \cap I_t(Y')| = |I_k(X) \cap I_t(Y)| + |I_{k-1}(X') \cap I_{t-1}(Y')|.$$

In this case the argument matching is easy, as $d_L(X',Y') = d_L(X,Y) = t-k+2\ell$. Using the induction hypothesis, we may write $|I_k(X) \cap I_t(Y)| = \mathcal{N}_2^+(n-1,t,k,\ell)$ and $|I_{k-1}(X') \cap I_{t-1}(Y')| = \mathcal{N}_2^+(n,t-1,k-1,\ell)$. We use Lemma 6 to conclude that

$$|I_k(X') \cap I_t(Y')| = \mathcal{N}_2^+(n-1,t,k,\ell) + \mathcal{N}_2^+(n,t-1,k-1,\ell) = \mathcal{N}_2^+(n,t,k,\ell),$$

and thus complete the proof. To retrieve the formula given by (11), take $t=k$ in (10). ∎

## IV. Maximum number of common supersequences: general case

Now we are ready to prove the main result of the present work, the general form of Theorem 5.

**Theorem 8.** *Let $n$ be a positive integer, $t, k, \ell$ be non-negative integers such that $t \geqslant k \geqslant \ell$ and $n \geqslant \ell$, and let $q$ be an integer with $q \geqslant 2$. Then,*

$$N_q^+(n,t,k,\ell) = \sum_{j=\ell}^{k} \sum_{i=0}^{k-j} \binom{t-k+2j}{j} \binom{t+j-i}{t-k+2j} \binom{n+t}{i} (q-1)^i (-1)^{k+j-i}. \tag{20}$$

*If $t = k$,*

$$N_q^+(n,t,\ell) = \sum_{j=\ell}^{t} \sum_{i=0}^{t-j} \binom{2j}{j} \binom{t+j-i}{2j} \binom{n+t}{i} (q-1)^i (-1)^{t+j-i}. \tag{21}$$

This section is organized as follows. First, we discuss important special cases of Theorem 8. In particular, we show how to recover Levenshtein's result for $N_q^+(n,t)$ from [27] as a special case. Afterwards, we provide the proof.

### A. Corollaries

Specific cases of Theorem 8 yield a variety of interesting results. Before we present these results, we require two auxiliary binomial identities. The purpose of these identities is to help simplify the more complex formulas in (20) and (21) for special cases. The proofs are found in the appendix.

**Lemma 9.**

1. *For $m \geqslant 0$,*

$$\sum_{j=0}^{m} \binom{2j}{j} \binom{m+j}{2j} (-1)^{m+j} = 1.$$

2. *For $n, m, t, j \geqslant 0$ and $t + j \geqslant m$,*

$$\sum_{i=0}^{m} \binom{t+j-i}{t+j-m} \binom{n+t}{i} (-1)^{m-i} = \binom{n+m-j-1}{m}.$$

We are now ready to proceed with our corollaries. We begin by showing that it is possible to recover Levenshtein's formula for $N_q^+(n,t) = \max_{X \neq Y \in \mathbb{F}_q^n} |I_t(X) \cap I_t(Y)|$ by taking $\ell = 1$ in (21). In other words, the maximum number of supersequences is attained by taking sequences at the smallest possible ($d_L = 2$) nonzero distance.

**Corollary 10 (Levenshtein's result for $N_q^+(n,t,\ell=1)$)** *For $n$ a positive integer and $t$ a non-negative integer,*

$$N_q^+(n,t,1) = N_q^+(n,t) = \sum_{i=0}^{t-1} \binom{n+t}{i} (q-1)^i (1-(-1)^{t-i}). \tag{22}$$

*Proof:* From the first identity in Lemma 9, we have that $\sum_{j=1}^{m} \binom{2j}{j} \binom{m+j}{2j} (-1)^{m+j} = 1 - (-1)^m$. Taking $m = t - i$ yields

$$\sum_{j=1}^{t-i} \binom{2j}{j} \binom{t-i+j}{2j} (-1)^{t-i+j} = 1 - (-1)^{t-i}. \tag{23}$$

If we exchange the sums in $i$ and $j$, we can rewrite the $\ell = 1$ case in (21) as

$$N_q^+(n,t,1) = \sum_{i=0}^{t-1} \sum_{j=1}^{t-i} \binom{2j}{j} \binom{t+j-i}{2j} \binom{n+t}{i} (q-1)^i (-1)^{t+j-i}. \tag{24}$$

Applying (23), we have that

$$N_q^+(n,t,1) = \sum_{i=0}^{t-1} \binom{n+t}{i} (q-1)^i (1-(-1)^{t-i}),$$

as desired. ∎

Note that we did not require the distance parameter $\ell$ to be positive in Theorem 8. In fact, $\ell = 0$ implies $d_L(X,Y) = 0$, or $X = Y$. In other words, all supersequences of $X$ are "common" supersequences (of $X$ and $X$), so we expect $N_q^+(n,t,0)$ to reduce to the formula for the number of supersequences $I_q(n,t)$. Happily, this is the case:

**Corollary 11 ($\ell = 0$ case)** *For $n$ a positive integer and $t$ a non-negative integer,*

$$N_q^+(n,t,0) = I_q(n,t) = \sum_{i=0}^{t} \binom{n+t}{i} (q-1)^i.$$

*Proof:* We exchange the sums for $i$ and $j$ in (21) with $\ell = 0$. This gives

$$N_q^+(n,t,0) = \sum_{i=0}^{t} \sum_{j=0}^{t-i} \binom{2j}{j} \binom{t+j-i}{2j} \binom{n+t}{i} (q-1)^i (-1)^{t+j-i}. \tag{25}$$

Now set $m = t - i$ in the first part of Lemma 9 and apply the result to (25). We get

$$N_q^+(n,t,0) = \sum_{i=0}^{t} \binom{n+t}{i} (q-1)^i,$$

which is just $I_q(n,t)$. ∎

The case $\ell = 0$ represents the minimum distance criterion. We also consider the maximum criterion. Recall that in Theorem 8 we required that $k \geqslant \ell$. What happens if $\ell > k$? In this case, the number of common supersequences is always 0. If a common supersequence $Z$ existed for $X \in \mathbb{F}_q^{n+t-k}$ and $Y \in \mathbb{F}_q^n$ with $d_L(X,Y) = 2\ell + t - k$, then, we could produce $Y$ from $X$ with $t + k$ insertions and deletions. However, since $\ell > k$, $t + k < 2\ell + t - k$, a contradiction. This is especially easy to see for equal-length sequences ($t = k$).

The maximum distance with a non-zero number of common supersequences is for $\ell = k$. In that case, the formula in Theorem 8 reduces to $\binom{t+k}{k}$. Here, we can even see a direct combinatorial interpretation of the formula. Consider for example $X = 00\ldots0$ and $Y = 11\ldots1$, where $X$ is made up of $t$ 0's and $Y$ is made up of $k$ 1's. Then, any common supersequence (formed by $t$ insertions into $Y$ and $k$ insertions into $X$) is a sequence with $t$ 0's and $k$ 1's. There are clearly $\binom{t+k}{k}$ such sequences.

Finally, we wish to reconcile the binary result (Theorem 5) with the non-binary result (Theorem 8). We note that these are not in the same form, so that taking $q = 2$ in Theorem 8 is not sufficient.

**Corollary 12 (Binary Case)** *Let $n$ be a positive integer and $t, k, \ell$ be non-negative integers such that $t \geqslant k \geqslant \ell$ and $n \geqslant \ell$. Then, we have the formula*

$$N_2^+(n,t,k,\ell) = \sum_{j=\ell}^{k} \binom{t-k+2j}{j} \binom{n+k-(2j+1)}{k-j}.$$

*Proof:* We take $m = k - j$ in the second identity of Lemma 9, giving

$$\sum_{i=0}^{k-j} \binom{t+j-i}{t+2j-k} \binom{n+t}{i} (-1)^{k-j-i} = \binom{n+k-2j-1}{k-j}.$$

Now, applying this result, we have

$$\begin{aligned} N_2^+(n,t,k,\ell) &= \sum_{j=\ell}^{k} \sum_{i=0}^{k-j} \binom{t-k+2j}{j} \binom{t+j-i}{t-k+2j} \binom{n+t}{i} (-1)^{k+j-i} \\ &= \sum_{j=\ell}^{k} \binom{t-k+2j}{j} \binom{n+k-2j-1}{k-j}, \end{aligned}$$

as desired. ∎

Next we are ready for a proof of Theorem 8.

### *B. Proof*

The proof of Theorem 8 uses the same approach as that of the binary version; however, the underlying recursions are more complex. We denote by $\mathcal{N}_q^+(n,t,k,\ell)$ the formula

$$\mathcal{N}_q^+(n,t,k,\ell) = \sum_{j=\ell}^{k} \sum_{i=0}^{k-j} \binom{t-k+2j}{j} \binom{t+j-i}{t-k+2j} \binom{n+t}{i} (q-1)^i (-1)^{k+j-i}.$$

Thus, it is again our goal to show that $N_q^+(n,t,k,\ell) = \mathcal{N}_q^+(n,t,k,\ell)$.

The formula $\mathcal{N}_q^+(n,t,k,\ell)$ satisfies two recursions.

**Lemma 13.** *For $n \geqslant 1$, $q \geqslant 2$ and $t, k, \ell \geqslant 1$ with $t \geqslant k \geqslant \ell$, $\mathcal{N}_q^+(n,t,k,\ell)$ satisfies the recursions*

$$\mathcal{N}_q^+(n,t,k,\ell) = \mathcal{N}_q^+(n-1,t,k,\ell) + (q-1)\mathcal{N}_q^+(n,t-1,k-1,\ell),$$

*and*

$$\mathcal{N}_q^+(n,t,k,\ell) = \mathcal{N}_q^+(n,t-1,k,\ell) + \mathcal{N}_q^+(n-1,t,k-1,\ell-1) + (q-2)\mathcal{N}_q^+(n,t-1,k-1,\ell).$$

Note that unlike in the binary case, the first recursion has a $(q-1)$ factor for the second term. The second recursion also contains a third term not found in the binary version. We defer the proof of Lemma 13 to the appendix. As in the binary case, we show that maximization $N_q^+(n,t,k,\ell)$ satisfies similar recursions:

**Lemma 14.** *Let $n$ and $q \geqslant 2$ be positive integers and $t, k, \ell$ be non-negative integers such that $t \geqslant k \geqslant \ell$. Let $X' = aX, Y' = bY$ be two sequences satisfying $X' \neq Y'$, $X' \in \mathbb{F}_q^{n+t-k}, Y' \in \mathbb{F}_q^n$, and $d_L(X',Y') = t-k+2\ell$. Then, if $a = b$,*

$$|I_k(X') \cap I_t(Y')| \leqslant N_q^+(n-1,t,k,\ell) + (q-1)N_q^+(n,t-1,k-1,\ell),$$

*and if $a \neq b$,*

$$|I_k(X') \cap I_t(Y')| \leqslant N_q^+(n,t-1,k,\ell) + N_q^+(n-1,t,k-1,\ell-1) + (q-2)N_q^+(n,t-1,k-1,\ell).$$

*Proof:* We are given $X', Y'$ satisfying $X' \neq Y'$, $X' \in \mathbb{F}_q^{n+t-k}, Y' \in \mathbb{F}_q^n$, and $d_L(X',Y') = t-k+2\ell$. We have $X' = aX$ and $Y' = bY$, with $a, b \in \mathbb{F}_q$. In the case $a = b$, from (6),

$$|I_k(aX) \cap I_t(bY)| = |I_k(aX) \cap I_t(aY)| = |I_k(X) \cap I_t(Y)| + (q-1)|I_{k-1}(aX) \cap I_{t-1}(aY)|. \tag{26}$$

The argument matching is identical to that in the proof of Lemma 7. We have that $d_L(X',Y') = t-k+2\ell$ and $X' = aX$, $Y' = aY$, so that $d_L(X,Y) = t-k+2\ell$. We have that $|I_k(X) \cap I_t(Y)| \leqslant N_q^+(n-1,t,k,\ell)$ and $|I_{k-1}(aX) \cap I_{t-1}(aY)| \leqslant N_q^+(n,t-1,k-1,\ell)$. Putting this into (26) gives

$$|I_k(aX) \cap I_t(bY)| \leqslant N_q^+(n-1,t,k,\ell) + (q-1)N_q^+(n,t-1,k-1,\ell).$$

The next case is $a \neq b$. Then, from (7),

$$|I_k(aX) \cap I_t(bY)| = |I_k(X) \cap I_{t-1}(bY)| + |I_{k-1}(aX) \cap I_t(Y)| + (q-2)|I_{k-1}(aX) \cap I_{t-1}(bY)|. \tag{27}$$

Since $d_L(aX,bY) = t-k+2\ell$, we have, from Claim 2, that $d_L(X,bY) \in \{t-k+2\ell-1, t-k+2\ell+1\}$ and $d_L(aX,Y) \in \{t-k+2\ell-1, t-k+2\ell+1\}$. Again, we bound the terms in (27) with formulas of the type $N_q^+(n,t,k,\ell)$. Using the same ideas as in the proof of Lemma 7, the argument tuples are given by $\{(n,t-1,k,\ell),(n,t-1,k,\ell+1)\}$ for the first term, $\{(n-1,t,k-1,\ell-1),(n-1,t,k-1,\ell)\}$ for the second term, and $(n,t-1,k-1,\ell)$ for the last term. Thus,

$$|I_k(X) \cap I_{t-1}(bY)| \leqslant \max\{N_q^+(n,t-1,k,\ell), N_q^+(n,t-1,k,\ell+1)\} = N_q^+(n,t-1,k,\ell),$$

$$|I_{k-1}(aX) \cap I_t(Y)| \leqslant \max\{N_q^+(n-1,t,k-1,\ell-1), N_q^+(n-1,t,k-1,\ell)\} = N_q^+(n-1,t,k-1,\ell-1),$$

and, finally,

$$|I_{k-1}(aX) \cap I_{t-1}(bY)| \leqslant N_q^+(n,t-1,k-1,\ell).$$

Plugging these inequalities into (27) yields

$$|I_k(aX) \cap I_t(bY)| \leqslant N_q^+(n,t-1,k,\ell) + N_q^+(n-1,t,k-1,\ell-1) + (q-2)N_q^+(n,t-1,k-1,\ell),$$

and we are done. ∎

We proceed with the proof of Theorem 8.

*Proof:* We first use induction on $n+t+k$ to show that

$$N_q^+(n,t,k,\ell) \;\leqslant\; \mathcal{N}_q^+(n,t,k,\ell). \tag{28}$$

The base cases are $n+t+k \in \{1,2\}$. First we consider $n \in \{1,2\}$ and $t = k = \ell = 0$. Since $I_0(X') \cap I_0(Y') \subseteq \{X'\}$, we have that $|I_0(X') \cap I_0(Y')| \leqslant 1$. The right hand side of (28) evaluates to $\binom{0}{0}\binom{0}{0}\binom{n+0}{0} = 1$, as we wished. The other possibility is $n = 1, t = 1, k = 0$, and $\ell = 0$. We have that $I_0(X') \cap I_1(Y') \subseteq \{X'\}$, and indeed, $\mathcal{N}_q^+(1,1,0,0) = \binom{1}{0}\binom{1}{1}\binom{2}{0} = 1$.

Assume that the claim in (28) holds for all $n+t+k < m$. We prove that it is true for $n+t+k = m$. Take sequences $X', Y'$, where $X' \neq Y'$, $X' \in \mathbb{F}_q^{n+t-k}, Y' \in \mathbb{F}_q^n$, and $d_L(X',Y') = t-k+2\ell$, and $n+t+k = m$. Write $X' = aX$ and $Y' = bY$, with $a, b \in \mathbb{F}_q$.

As before, we look at the first symbol. The first case is that $X'$ and $Y'$ start with the same symbol, so that $a = b$. Then, using Lemma 14, the induction hypothesis, and the first recursion in Lemma 13, we write

$$|I_k(aX) \cap I_t(bY)| \leqslant N_q^+(n-1,t,k,\ell) + (q-1)N_q^+(n,t-1,k-1,\ell)$$

$$\leqslant \mathcal{N}_q^+(n-1,t,k,\ell) + (q-1)\mathcal{N}_q^+(n,t-1,k-1,\ell)$$
$$= \mathcal{N}_q^+(n,t,k,\ell).$$

The other case is $a \neq b$, so that $X'$ and $Y'$ start with different symbols. Now we apply the second result in Lemma 14, the induction hypothesis, and the second recursion in Lemma 13, yielding

$$\begin{aligned}|I_k(aX) \cap I_t(bY)| &\leqslant N_q^+(n,t-1,k,\ell) + N_q^+(n-1,t,k-1,\ell-1) + (q-2)N_q^+(n,t-1,k-1,\ell)\\ &\leqslant \mathcal{N}_q^+(n,t-1,k,\ell) + \mathcal{N}_q^+(n-1,t,k-1,\ell-1) + (q-2)\mathcal{N}_q^+(n,t-1,k-1,\ell)\\ &= \mathcal{N}_q^+(n,t,k,\ell).\end{aligned}$$

Thus, $N_q^+(n,t,k,\ell) \leqslant \mathcal{N}_q^+(n,t,k,\ell)$. Now we show that there exist sequences $X'$,$Y'$ that attain the maximum; that is, we find $X', Y'$ with $|I_k(X') \cap I_t(Y')| = \mathcal{N}_q^+(n,t,k,\ell)$. This allows us to conclude that $N_q^+(n,t,k,\ell) = \mathcal{N}_q^+(n,t,k,\ell)$, completing the proof.

Just as in the binary case, we select the sequences

$$X' = \underbrace{00\ldots0}_{t-k+n\ 0\text{'s}} \text{ and } Y' = \underbrace{11\ldots1}_{\ell\ 1\text{'s}}\underbrace{00\ldots0}_{n-\ell\ 0\text{'s}}.$$

Note of course that we could have selected any sequences with the structure $X' = \underbrace{aa\ldots a}_{t-k+n\ a\text{'s}}$ and $Y' = \underbrace{bb\ldots b}_{\ell\ b\text{'s}}\underbrace{aa\ldots a}_{n-\ell\ a\text{'s}}$ for $a \neq b$ and $a, b \in \mathbb{F}_q$.

As before, the induction is on $n+t+k$. The base cases are $n+t+k \in \{1,2\}$. If $n \in \{1,2\}$ and $t = k = \ell = 0$, $X' = 00$ and $Y' = 00$ or $X' = 0$ and $Y' = 0$. Thus, $|I_0(X') \cap I_0(Y')| = 1 = \mathcal{N}_q^+(n,0,0,0)$, as desired. If $n = t = 1$ and $k = \ell = 0$, $X' = 00$ and $Y' = 0$. Then $|I_0(X') \cap I_1(Y')| = 1 = \mathcal{N}_q^+(1,1,0,0)$.

Assume that the induction hypothesis holds for $n+t+k < m$; we show it is true for $n+t+k = m$. If $\ell \geqslant 1$, we apply (7) to write $|I_k(X') \cap I_t(Y')| = |I_k(X) \cap I_{t-1}(Y')| + |I_{k-1}(X') \cap I_t(Y)| + (q-2)|I_{k-1}(X') \cap I_{t-1}(Y')|$, where $X = \underbrace{00\ldots0}_{t-k+n-1\ 0\text{'s}}$ and $Y = \underbrace{11\ldots1}_{\ell-1\ 1\text{'s}}\underbrace{00\ldots0}_{n-\ell\ 0\text{'s}}$. The argument matching is the same as in the proof of the binary version of the theorem; we do not repeat it here. The result, using the induction hypothesis and Lemma 13 is

$$|I_k(X') \cap I_t(Y')| = \mathcal{N}_q^+(n,t-1,k,\ell) + \mathcal{N}_q^+(n-1,t,k-1,\ell-1) + (q-2)\mathcal{N}_q^+(n,t-1,k-1,\ell) = \mathcal{N}_2^+(n,t,k,\ell).$$

If $\ell = 0$, $X'$ and $Y'$ both start with 0 so we apply (6) to write $|I_k(X') \cap I_t(Y')| = |I_k(X) \cap I_t(Y)| + (q-1)|I_{k-1}(X') \cap I_{t-1}(Y')|$. In this case too we apply the induction hypothesis and Lemma 13, giving

$$|I_k(X') \cap I_t(Y')| = \mathcal{N}_q^+(n-1,t,k,\ell) + (q-1)\mathcal{N}_q^+(n,t-1,k-1,\ell) = \mathcal{N}_q^+(n,t,k,\ell),$$

and thus completing the proof. To retrieve the formula given by (21), take $t = k$ in (20). ■

## V. Reconstruction in Varshamov-Tenengolts (VT) Codes

Thus far, we have examined the number of traces required to distinguish between (sufficiently distant) sequences. We answered this question by deriving the $N_q^+(n,t,k,\ell)$ formula. However, this expression represents the worst case. We may wonder whether in some *particular* code the situation is better. That is, we may ask whether the codewords of a code require fewer than $N_q^+(n,t,k,\ell)+1$ traces for reconstruction.

One major challenge when dealing with such a question is that there are not very many explicit codes correcting a fixed number of insertions and deletions. We will examine the most famous such code, the binary single insertion/deletion-correcting Varshmov-Tennenegolts (VT) code [25], [29]:

$$\mathcal{C}_{VT}(n,a) := \{X = (x_1,\ldots,x_n) \in \mathbb{F}_2^n : \sum_{i=1}^{n} i \cdot x_i \equiv a \bmod n+1\}.$$

We may take any $0 \leqslant a \leqslant n$ to form a different code. The VT codes partition the space $\{0,1\}^n$. Since the VT codes correct a single insertion or deletion and have equal-length codewords, for any $E, F \in \mathcal{C}_{VT}(n,a)$ $(E \neq F)$, we have that $d_L(E,F) \geqslant 4$. This represents the case of $\ell = 2$ in our notation. Thus we seek to find out whether there exist (unordered) pairs of codewords $\{E,F\}$ with $E, F \in \mathcal{C}_{VT}(n,a)$ for some particular $a$ such that $|I_t(E) \cap I_t(F)| = N_q^+(n,t,2)$. If such pairs exist for each $a$, we can conclude that the VT codes require the worst-case $N_2^+(n,t,2)+1$ traces for reconstruction (and we are, perhaps, motivated to seek codes with similar properties to the VT codes but with fewer traces needed for reconstruction.)

In this section, we prove an even stronger result. Not only is there always a pair of codewords achieving the worst case, but there are exponentially many such pairs for each VT code. In particular, we prove that for any $n \geqslant 4$ and any $a$ with $0 \leqslant a \leqslant n$,

there exists a set $S_a$ of unordered pairs of elements, where for any element $\{E, F\} \in S_a$, we have $E, F \in \mathcal{C}_{VT}(n, a)$, $E \neq F$, $N_2^+(n, t, 2) = |I_t(E) \cap I_t(F)|$ and

$$|S_a| \geqslant 2^{n - \lceil \log_2(n) \rceil - 3}.$$

We first establish some simple claims. We make use of the following notation. Let $V = (v_1, \ldots, v_{n-2}) \in \mathbb{F}_2^{n-2}$ be a sequence of length $n-2$. We write

$$X(n, p, V) := (v_1, \ldots, v_{p-1}, 1, 1, v_p, \ldots, v_{n-2}) \in \mathbb{F}_2^n,$$

i.e., $X(n, p, V)$ is a sequence whose components are equal to $V$ in the first $p-1$ positions, followed by $1, 1$, then by the remaining $n-p-1$ bits in $V$. Similarly, let

$$Z(n, p, V) = (v_1, \ldots, v_{p-1}, 0, 0, v_p, \ldots, v_{n-2}) \in \mathbb{F}_2^n,$$
$$Y(n, p, V) = (v_1, \ldots, v_{p-1}, 0, v_p, 0, v_{p+1}, \ldots, v_n) \in \mathbb{F}_2^n,$$

and

$$W(n, p, V) = (v_1, \ldots, v_{p-1} 1, v_p, 1, v_{p+1}, \ldots, v_n) \in \mathbb{F}_2^n.$$

Before continuing we provide a small example that illustrates the main ideas.

**Example 1.** We take $V = 100100 \in \mathbb{F}_2^6$. Then, we have that $X(8, 4, V) = 10011100$, $Z(8, 4, V) = 10000100$. Note that $X(8, 4, V)$ and $Z(8, 4, V)$ are in the same VT code, $\mathcal{C}_{VT}(8, 7)$, since $\sum_{i=1}^8 i x_i = 1 + 4 + 5 + 6 \equiv 7 \bmod 9$ and $\sum_{i=1}^8 i z_i = 1 + 6 \equiv 7 \bmod 9$.

Note that we always have that $d_L(X(n, p, V), Z(n, p, V)) = 4$, since the two sequences are the same except for the $0, 0$ and $1, 1$ bits in the middle. Similarly, we have that $d_L(Y(n, p, V), W(n, p, V)) = 4$. Next we show that if we let $p$ be the central position in some $V$, then the resulting sequences $X$ and $Z$ satisfy $|I_t(X) \cap I_t(Z)| = N_2^+(n, t, \ell = 2)$ and similarly $|I_t(Y) \cap I_t(W)| = N_2^+(n, t, \ell = 2)$.

**Lemma 15.** *For any $n \geqslant 4$, $t \geqslant 2$, and $V \in \mathbb{F}_2^{n-2}$,*

$$N_2^+(n, t, 2) = \left| I_t(X(n, \left\lfloor \frac{n}{2} \right\rfloor, V)) \cap I_t(Z(n, \left\lfloor \frac{n}{2} \right\rfloor, V)) \right| = \left| I_t(Y(n, \left\lfloor \frac{n}{2} \right\rfloor, V)) \cap I_t(W(n, \left\lfloor \frac{n}{2} \right\rfloor, V)) \right|.$$

*Proof:* We show that $N_2^+(n, t, 2) = |I_t(X(n, \lfloor \frac{n}{2} \rfloor, V)) \cap I_t(Z(n, \lfloor \frac{n}{2} \rfloor, V))|$. Just as in our earlier proofs, we use induction; this time on $n + t$. Since $n \geqslant 4$ and $t \geqslant 2$, we have $n + t = 6$ as our base case. It can be verified exhaustively that for any $V \in \mathbb{F}_2^2$, $N_2^+(4, 2, 2) = |I_2(X(4, 2, V)) \cap I_2(Z(4, 2, V))| = \sum_{j=\ell}^t \binom{2j}{j} \binom{n+t-(2j+1)}{t-j} = \binom{4}{2}\binom{10-5}{2-2} = 6$, as desired.

Suppose that $N_2^+(n, t, 2) = |I_t(X(n, \lfloor \frac{n}{2} \rfloor, V)) \cap I_t(Z(n, \lfloor \frac{n}{2} \rfloor, V))|$ for all $n + t < m$ and consider the case where $n + t = m$. Suppose that $n$ is even. The case where $n$ is odd can be proven using similar arguments. Let $V' \in \mathbb{F}_2^{n-3}$ be the sequence obtained by deleting the first bit from $V$. Since $X(n, \frac{n}{2}, V)$ and $Z(n, \frac{n}{2}, V)$ start with the same bit, we can use (8) in Claim 4 to write

$$\begin{aligned} \left| I_t(X(n, \frac{n}{2}, V)) \cap I_t(Z(n, \frac{n}{2}, V)) \right| &= \left| I_t(X(n-1, \left\lfloor \frac{n-1}{2} \right\rfloor, V')) \cap I_t(Z(n-1, \left\lfloor \frac{n-1}{2} \right\rfloor, V')) \right| \\ &\quad + \left| I_{t-1}(X(n, \frac{n}{2}, V)) \cap I_{t-1}(Z(n, \frac{n}{2}, V)) \right|. \end{aligned}$$

Applying the inductive hypothesis, $|I_t(X(n-1, \lfloor \frac{n-1}{2} \rfloor, V')) \cap I_t(Z(n-1, \lfloor \frac{n-1}{2} \rfloor, V'))| = N_2^+(n-1, t, 2)$ and $|I_{t-1}(X(n, \frac{n}{2}, V)) \cap I_{t-1}(Z(n, \frac{n}{2}, V))| = N_2^+(n, t-1, 2)$. From Lemma 6, $N_2^+(n-1, t, 2) + N_2^+(n, t-1, 2) = N_2^+(n, t, 2)$ so that $N_2^+(n, t, 2) = |I_t(X(n, \lfloor \frac{n}{2} \rfloor, V)) \cap I_t(Z(n, \lfloor \frac{n}{2} \rfloor, V))|$ as desired. The expression $N_2^+(n, t, 2) = |I_t(Y(n, \lfloor \frac{n}{2} \rfloor, V)) \cap I_t(W(n, \lfloor \frac{n}{2} \rfloor, V))|$ can be proven using nearly identical logic. ∎

Our eventual goal is to find codeword pairs that are in a particular VT code and achieve the worst-case number of common supersequences. These pairs must satisfy the checksum constraint that defines the VT code. To ensure this, we will need to control certain positions in these codewords. We make use of a function $FP$ that takes as an argument an integer $n$ and returns a subset of integers (related to the positions that we will control) of size at most $\lceil \log(n) \rceil + 1$. The set returned by the function $FP$ is defined iteratively as follows:

1) Initialize $T = \{1\}$.
2) Let $t = \sum_{i \in T} i$.
   a) If $t \geqslant n$, then define $FP(n) = T$ and stop.
   b) If $n$ is even and $t + 1 \in \{\frac{n}{2}, \frac{n}{2} + 1\}$, then let $T = T \cup \{\frac{n}{2} - 1\}$ and go back to step 2).
   c) If $n$ is odd and $t + 1 = \lfloor \frac{n}{2} \rfloor$, then let $T = T \cup \{\lfloor \frac{n}{2} \rfloor - 1\}$ and go back to step 2). If $n$ is odd and $t + 1 = \lfloor \frac{n}{2} \rfloor + 2$, then let $T = T \cup \{\lfloor \frac{n}{2} \rfloor + 1\}$ and go back to step 2).

d) Set $T = T \cup \{t+1\}$, and go back to step 2).

We illustrate the previous procedure via an example.

**Example 2.** Suppose $n = 16$. Then, after step 1) of the procedure to compute $FP(16)$, we have $T = \{1\}$. Next, $t = 1$ and we go to step 2-d) and get $T = \{1, 2\}$. Afterwards, we again go to step 2-d) and have $T = \{1, 2, 4\}$. At this point $t = 7$ so that $t + 1 = \frac{n}{2}$. Then from step 2-b), $T = \{1, 2, 4, 7\}$. Next, $t = 14$ and we go to step 2-d) again so that $T = \{1, 2, 4, 7, 15\}$. Finally, we reach step 2-a), and the procedure stops so that $FP(16) = T = \{1, 2, 4, 7, 15\}$. Notice that $|FP(16)| = \log_2(16) + 1 = 5$.

The idea of the algorithm producing the output of $FP(n)$ is to include positions that are roughly powers of 2 while avoiding certain central positions based on the parity of $n$. The reason for the avoidance is that sequences such as $X(n, \lfloor \frac{n}{2} \rfloor, V')$ have these positions already fixed and we cannot therefore control them in order to ensure that the output sequence is in a particular VT code. The remaining positions, however, form a basis, (that is, a linear combination of them produces any $a$ with $0 \leqslant a \leqslant n$ modulo $n+1$) so that we can control their output in the checksum to fix the $a$ VT code parameter.

We can conclude that,

**Lemma 16.** *For any $n \geqslant 4$, and integer $m \leqslant n$, there exists a subset $T' \subseteq FP(n)$ where $\sum_{i \in T'} i = m$. In addition, $\lceil \log_2(n) \rceil \leqslant |FP(n)| \leqslant \lceil \log_2(n) \rceil + 1$. Furthermore, if $n$ is even, we have $\{\frac{n}{2}, \frac{n}{2} + 1\} \notin FP(n)$ and if $n$ is odd, then $\{\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor + 2\} \notin FP(n)$.*

The idea in Lemma 16 is that a linear combination of at most $\lceil \log_2(n) \rceil$ numbers in $[n]$ can generate any $m$ for $0 \leqslant m \leqslant n$. These numbers are precisely those returned by the function $FP(n)$. Therefore, we can force a sequence of length $n$ to be in any particular VT code by controlling the sequence components at these positions. The subset $T'$ represents those positions where we will place a 1 in the codeword, while the positions given by $FP(n) \setminus T'$ will be set to 0. The idea is explained in further detail in the following proof of the main result of this section.

**Theorem 17.** *For any $n \geqslant 4$ and $a \in \mathbb{Z}_{n+1}$, there exists a set of unordered pairs $S_a$ where for any pair $\{E, F\} \in S_a$, we have $E, F \in \mathcal{C}_{VT}(n, a)$, $E \neq F$, $N_2^+(n, t, 2) = |I_t(E) \cap I_t(F)|$, and*

$$|S_a| \geqslant 2^{n - \lceil \log_2(n) \rceil - 3}.$$

*Proof:* We start by counting the number of different ways to form a pair $\{E, F\} \in S_a$. First, consider the case where $n$ is even. Let $U = FP(n) \cup \frac{n}{2} \cup (\frac{n}{2} + 1)$ be a set of positions in our codewords. We can select the remaining positions, that is, $[n] \setminus U$, freely and still form a codeword. Therefore, we have $2^{n - |FP(n)| - 2}$ choices for these positions. Next, let us say that $\sum_{k \in [n] \setminus U} k e_k \equiv c \bmod n+1$. Then, using Lemma 16, we fix the components of $E$ at positions in the set $FP(n)$ so that $\sum_{k \in FP(n)} k e_k \equiv a - c \bmod n+1$. Finally let $(e_{\frac{n}{2}}, e_{\frac{n}{2}+1}) = (0, 0)$. (Thus, we may write $E = Z(n, \frac{n}{2}, E')$ for some vector $E' \in \mathbb{F}_2^{n-2}$.) Notice that $E \in \mathcal{C}_{VT}(n, a)$ since $\sum_{k \in [n]} k e_k = \sum_{k \in [n] \setminus U} k e_k + \sum_{k \in FP(n)} k e_k + \frac{n}{2} e_{\frac{n}{2}} + (\frac{n}{2} + 1) e_{\frac{n}{2}+1} \equiv a \bmod n+1$ as desired.

Let $F = (e_1, \ldots, e_{\frac{n}{2}-1}, 1, 1, e_{\frac{n}{2}+2}, \ldots, e_n) = (f_1, \ldots, f_n)$, so that $F = X(n, \frac{n}{2}, E')$. Then, $F \in \mathcal{C}_{VT}(n, a)$, since $\sum_{k \in [n]} k f_k = \sum_{k \in [n]} k e_k + \frac{n}{2} + (\frac{n}{2} + 1) \equiv a \bmod (n+1)$. Then, from Lemma 15, $|I_t(E) \cap I_t(F)| = N_2^+(n, t, 2)$. Thus, $|S_a| \geqslant 2^{n - |FP(n)| - 2} \geqslant 2^{n - \lceil \log_2(n) \rceil - 3}$.

Next we examine the case of $n$ odd. We proceed in the same manner as before except that we first select the values of $E$ except in positions from the set $U = FP(n) \cup \lfloor \frac{n}{2} \rfloor \cup (\lfloor \frac{n}{2} \rfloor + 2)$. Afterwards we assign values to components in $E$ whose indices belong to set $FP(n)$ in such a way that $E \in \mathcal{C}_{VT}(n, a)$. Finally $e_{\lfloor \frac{n}{2} \rfloor}, e_{\lfloor \frac{n}{2} \rfloor + 2}$ are both set to zero so that $E = Y(n, \lfloor \frac{n}{2} \rfloor, E')$. Next, $F = (f_1, \ldots, f_n)$ is set to be equal to $E$ except that $f_{\lfloor \frac{n}{2} \rfloor} = f_{\lfloor \frac{n}{2} \rfloor + 2} = 1$. Thus $F = W(n, \lfloor \frac{n}{2} \rfloor, E')$. Using the same arguments as in the previous paragraph, it can be shown that $E, F \in \mathcal{C}_{VT}(n, a)$. Furthermore, from Lemma 15, $|I_t(E) \cap I_t(F)| = N_2^+(n, t, 2)$ and thus $|S_a| \geqslant 2^{n - \lceil \log_2(n) \rceil - 3}$. ∎

## VI. Other Channels

Thus far we have only been concerned with insertion channels. It is reasonable to ask what occurs in the cases of deletion or mixed insertion/deletion channels. It is not too surprising that finding expressions for similar problems for these channels is much harder: the deletion channel is much less symmetric compared to the insertion channel, and the insertion/deletion channel deals with the challenges of both. We provide a few specific results while leaving the general questions open for further study.

### A. Deletion Channel

Levenshtein examined exact reconstruction for deletion channels in [27]. He defined $N_q^-(n,t) := \max_{X,Z\in\mathbb{F}_q^n, X\neq Z} |D_t(X)\cap D_t(Z)|$ and showed that

$$N_q^-(n,t) = \sum_{i=1}^{q-1} D_q(n-i-1,t-i) + D_q(n-2,t-1).$$

Here, $D_q(n,t)$ is the maximum size of the deletion ball $D_t(X)$ for some $X\in\mathbb{F}_q^n$. It is known that $D_q(n,t)$ satisfies the recursion $D_q(n,t) = \sum_{i=0}^{t}\binom{n-t}{i} D_{q-1}(t,t-i)$, where $D_1(n,t)=1$ if $n\geqslant t\geqslant 0$ and $D_q(n,t)=0$ otherwise [33]. It is not hard to see that $D_2(n,t) = \sum_{i=0}^{t}\binom{n-t}{i}$. This enables us to write that the maximum number of common subsequences in the binary case is given by

$$N_2^-(n,t) = 2\sum_{i=0}^{t-1}\binom{n-t-1}{i}.$$

Just as before, we may ask what happens to the number of sequences required for reconstruction if we select the original sequences from an insertion/deletion-correcting code. We can analogously define

$$N_q^-(n,t,\ell) = \max_{\substack{X,Z\in\mathbb{F}_q^n \\ d_L(X,Z)\geqslant 2\ell}} |D_t(X)\cap D_t(Z)|.$$

Few results are known regarding $N_q^-(n,t,\ell)$. The work [32] is dedicated to the $\ell=2, q=2$ case (corresponding to VT codes). The authors showed that for $t\leqslant n/2$,

$$N_2^-(n,t,2) = 2D_2(n-4,t-2) + 2D_2(n-5,t-2) + 2D_2(n-7,t-2) + D_2(n-6,t-3) + D_2(n-7,t-3).$$

Our contribution consists of removing the reliance on recursions from this formula, yielding the exact expression

$$\begin{aligned} N_2^-(n,t,2) &= 2D_2(n-2,t-1) - 2\binom{n-t-3}{t-1} - \binom{n-t-4}{t-3} - \binom{n-t-5}{t-3} \\ &= 2\sum_{i=0}^{t-1}\binom{n-t-1}{i} - 2\binom{n-t-3}{t-1} - \binom{n-t-4}{t-3} - \binom{n-t-5}{t-3}. \end{aligned}$$

The proof is an easy induction.

### B. Insertion/Deletion Channel

What about the case of insertion/deletion channels? In general, this problem is quite hard, since even the sizes of $t$-insertion/$t$-deletion balls are not known beyond trivial cases. Let us slightly abuse notation as follows: given a set $S\subseteq\mathbb{F}_q^n$, we write $I_t(S)$ and $D_t(S)$ for $\cup_{X\in S} I_t(X)$ and $\cup_{X\in S} D_t(X)$, respectively. Then, the $t$-insertion/$t$-deletion ball centered $X$ may be written $B_t(X) := I_t(D_t(X))$.

Since the general version of the problem seems intractable, in this subsection we focus on providing a lower bound on the number of distinct distorted sequences (resulting from an insertion/deletion channel) required to reconstruct a binary sequence $X$. Specifically, we are interested in a lower bound on $N_H(\mathbb{F}_2^n, 2t)$, where $H$ is the set of single symbol insertions and deletions. (Note that the $2t$ argument refers to $t$ insertions and $t$ deletions.) We can write, in general, that

$$N_H(\mathbb{F}_2^n, 2t) = \max_{\substack{X,Z\in\mathbb{F}_2^n \\ X\neq Z}} |I_t(D_t(X))\cap I_t(D_t(Z))|.$$

We provide a lower bound on $N_H(\mathbb{F}_2^n, 2t)$ by computing the number of common distorted sequences in one particular (and non-trivial) case. This is the case of the so-called binary circular string $C_n = \underbrace{0101\ldots}_{n \text{ bits}}$ (or $\underbrace{10101\ldots}_{n \text{ bits}}$.) This string is particularly interesting; in [33] it is shown that[4]

$$C_n = \arg\max_{X\in\mathbb{F}_2^n} |D_t(X)|.$$

We begin by evaluating the size of the neighborhood (ball) centered at $C_n$, $B_t(C_n) = I_t(D_t(C_n))$.

**Theorem 18.** *The size of the neighborhood of the binary circular string $C_n\in\mathbb{F}_2^n$ is given by*

$$|B_t(C_n)| = |I_t(D_t(C_n))| = \sum_{i=0}^{2t}\binom{n}{i}. \tag{29}$$

[4]There are no expressions for $|D_t(X)|$ for general $t$; however, the minimal, maximal, and average values are known. The tightest known bounds on $|D_t(X)|$ are found in [18].

Before we proceed with the proof of Theorem 18, we comment on this result. S ince $C_n$ is known to maximize the deletion ball size $|D_t(X)|$, we may ask whether the string $C_n$ also maximizes the insertion/deletion neighborhood size. Surprisingly, this is not the case. Although $|B_t(C_n)|$ is quite large and in some small cases is in fact maximal, the string $X = 00110011\ldots$ generally has a larger degree. More details on which strings maximize the neighborhood size can be found in [34].

We use the following lemma in the proof of Theorem 18:

**Lemma 19.** *Let $n, t$ be positive integers with $n \geqslant 2t$. Let $C_{n-2t}$ be the substring formed by the first $n-2t$ bits of the circular string $C_n$. Then, the $t$-deletion ball centered at $C_n$ is exactly the $t$-insertion ball centered at $C_{n-2t}$. That is,*

$$D_t(C_n) = I_t(C_{n-2t}). \tag{30}$$

*Proof:* First, observe that $C_{n-2t}$ begins and ends with the same bit as $C_n$. We will show the result by induction on $n$.

The base case is $n = 2t$. Here, $C_{n-2t}$ is the empty string, and the right hand side in (30) is just $\mathbb{F}_2^t$, the set of all binary sequences of length $t$. It is easy to see that this set is equal to $D_t(C_n) = D_t(C_{2t}) = D_t(01\ldots01)$ (or $D_t(10\ldots10)$). We may delete either the 0 or the 1 in all of the $t$ consecutive 01 (or 10) pairs in order to produce any sequence of length $t$. This establishes the base case.

Now, we assume that $D_{t'}(C_m) = I_{t'}(C_{m-2t})$ for all $t' \leqslant t$ and $m$ satisfying $2t \leqslant m \leqslant n$. (The cases of $t' < t$ follow from the $t$ case by deleting and inserting identical elements.) We examine $D_t(C_{n+1})$ with the goal of showing that it is identical to $I_t(C_{n+1-2t})$. We take the last bit of $C_{n+1}$ (and thus, of $C_{n+1-2t}$) to be 1, without loss of generality. Consider some $X \in D_t(C_{n+1})$ so that $X$ ends in exactly $k$ consecutive 0s, with $0 \leqslant k \leqslant t$. We show that $X \in I_t(C_{n+1-2t})$.

If $k = 0$, $X$ ends in 1, like $C_{n+1}$. In this case, $X = Y1$ where $Y$ has length $n-t$. Then, $Y$ may be produced by $t$ deletions in the string $C_n$, which is itself the first $n$ bits of $C_{n+1}$. Thus, $Y \in D_t(C_n)$. By the induction hypothesis, $D_t(C_n) = I_t(C_{n-2t})$, so $Y \in I_t(C_{n-2t})$. Then, $Y$ can be produced by $t$ insertions to $C_{n-2t}$, so, since $C_{n-2t+1}$ ends in 1, indeed $X \in I_t(C_{n-2t+1})$.

If $0 < k \leqslant t$, $X$ ends with the substring $1\underbrace{00\ldots0}_{k\text{ 0's}}$. In fact, we may write $X = Y\underbrace{00\ldots0}_{k\text{ 0's}}$ for some string $Y$ of length $(n+1-t-k)$. $X$ results from the deletion of the last $k$ 1's from $C_{n+1}$, and the deletion of an additional $t-k$ elements from the first $n+1-2k$ bits of $C_{n+1}$, which themselves form $C_{n+1-2k}$. That is, $Y \in D_{t-k}(C_{n+1-2k})$. Applying the induction hypothesis, $Y$ is in the set $I_{t-k}(C_{n+1-2k-2(t-k)}) = I_{t-k}(C_{n+1-2t})$. Then, clearly $X \in I_t(C_{n+1-2t})$, as we may use the remaining $k$ insertions to add $k$ 0s to the end of $Y$ to produce $X$. We conclude that $D_t(C_{n+1}) \subseteq I_t(C_{n+1-2t})$.

The other direction is essentially identical. Take $Z \in I_t(C_{n+1-2t})$. If $Z$ ends in 1, then $Z = Y1$ where $Y$ may be formed by $t$ insertions into $C_{n-2t}$. By the induction hypothesis, $Y \in D_t(C_n)$, and since $C_{n+1}$ ends in 1, we have that $Z \in D_t(C_{n+1})$. If $Z$ ends in exactly $k$ 0s, $(1 \leqslant k \leqslant t)$ then $Z = Y\underbrace{00\ldots0}_{k\text{ 0's}}$ for some $Y$ of length $n+1-t-k$. Then, $Y$ can be formed by $t-k$ insertions into $C_{n+1-2t}$. By the induction hypothesis, $Y \in D_{t-k}(C_{n+1-2t+2(t-k)}) = D_{t-k}(C_{n+1-2k})$. Then, $Z \in D_t(C_{n+1})$, since we may use the remaining $k$ deletions to delete the last $k$ 1s in $C_{n+1}$. With this, $I_t(C_{n-2t+1}) \subseteq D_t(C_{n+1})$.

Thus, $D_t(C_{n+1}) = I_t(C_{n-2t+1})$, and we are done. ∎

Theorem 18 follows almost immediately from Lemma 19:

*Proof:* Let $n \geqslant 2t$. According to Lemma 19, $D_t(C_n) = I_t(C_{n-2t})$. Then, we have that

$$\begin{aligned}|B_t(C_n)| &= |\cup_{X\in D_t(C_n)} I_t(X)| = |\cup_{X\in I_t(C_{n-2t})} I_t(X)| \\ &= |I_{2t}(C_{n-2t})| = \sum_{i=0}^{2t}\binom{n}{i},\end{aligned}$$

where in the last step, we used the formula for the number supersequences formed by $2t$ insertions (3). The remaining cases for $n < 2t$ are identical to the base case $n = 2t$ in the proof of Lemma 19. Here too, $D_t(C_n) = F_2^{n-t}$, so that $\cup_{X\in D_t(C_n)} I_t(X) = \cup_{X\in F_2^{n-t}} I_t(X)$, implying that $|B_t(C_n)| = 2^n$. ∎

Theorem 18 is interesting, as in general it is very difficult to compute the exact ball size $|B_t(X)|$ for any non-trivial $X$ (such as any sequence that is not made up of all 0's or all 1's) or $t > 1$. The underlying symmetries for the circular string enable us to give this exact expression. We remark that Lemma 19 also yields an alternative way to compute the size of $D_t(C_n)$ [33].

Now we return to the problem of common distorted sequences. Recall that we are interested in computing $|I_t(D_t(X)) \cap I_t(D_t(Z))|$ for at least some non-trivial $X, Z \in \mathbb{F}_2^n$. Let us take $X = C_n = 10101\ldots$ and $Z = C'_n = 010101\ldots$. Note that $d_L(C_n, C'_n) = 2$, since we need only take the leading 1 in $C_n$ and move it to the end to reproduce $Z$. Now, we have that

$$\begin{aligned}&|I_t(D_t(C_n)) \cap I_t(D_t(C'_n))| \\ &\quad= |I_t(I_t(C_{n-2t})) \cap I_t(I_t(C'_{n-2t}))| \\ &\quad= |I_{2t}(C_{n-2t}) \cap I_{2t}(C'_{n-2t})| \\ &\quad= N_2^+(n-2t, 2t, 1)\end{aligned}$$

$$= 2 \sum_{i=0}^{2t-1} \binom{n}{i}.$$

The equality (rather than inequality) in the third step is easy to check. Therefore, we have our desired bound on $N_H(\mathbb{F}_2^n, 2t)$:

$$N_H(\mathbb{F}_2^n, 2t) \geqslant 2 \sum_{i=0}^{2t-1} \binom{n}{i}.$$

The important idea here is to replace deletions in our insertion/deletion channel with insertions. This idea is often useful when computing sizes of insertion/deletion balls, since deletions are much more difficult to deal with. Note that we can use a similar idea to compute the number of common distorted sequences for some other cases. For example, if we let $Z = C_n$, but take $X = 00\ldots0$, we have that $I_t(D_t(0\ldots0)) = I_t(0\ldots0) = I_2(n-t,t)$, which yields $|I_t(D_t(0\ldots0)) \cap I_t(D_t(C_n))| = |I_t(0\ldots0) \cap I_{2t}(C_{n-2t})| \leqslant N_2^+(n-2t, 2t, t, \frac{1}{2}(\lfloor \frac{n-2t}{2} \rfloor - t))$. A number of other similar expressions can be computed.

## VII. Conclusion

In this work, we examined the exact reconstruction of sequences that are codewords of synchronization (insertion/deletion-correcting) codes from traces that are the result of an insertion channel. We provided exact formulas for the number of traces necessary for the binary and non-binary cases of this problem (and additionally for the more exotic case where the codewords are of differing lengths.) These formulas resolve a problem left open by Levenshtein, who derived the first expressions for the uncoded case. We also examined traces produced by other channels, such as the insertion and deletion channel.

The expressions we found represent the worst-case number of traces needed when performing reconstruction in any code with the required minimum edit distance. We asked whether selecting a particular code allows us to reconstruct with fewer traces compared to the worst-case. We showed that for the popular single insertion/deletion-correcting Varshamov-Tenengolts codes, there are always many codeword pairs that require the worst-case number of traces for reconstruction. This inspires us to ask whether we can construct new codes that have similar properties to the VT codes, but better (smaller) requirements for reconstruction.

Our results can be viewed as a promising first step towards a more general theory for coded data reconstruction. This is a rich area with many interesting further questions. We are particularly interested in the equivalent problems for the cases of deletion channels (for $\ell > 2$) and combined insertions/deletions/substitutions channels, which accurately model real-life data reconstruction scenarios. Equally intriguing is a study of efficient algorithms for reconstruction given the necessary number of traces: for example, given $N_q^+(n,t,k,\ell)+1$ traces of $X$, what is the most efficient algorithm to reproduce $X$?

## References


[1] T. Batu, S. Kannan, S. Khanna, and A. McGregor, "Reconstructing strings from random traces," in *Proc. ACM-SIAM Symp. Discrete Algorithms*, pp. 910-918, 2004.

[2] T. Holenstein, M. Mitzenmacher, R. Panigrahy, and U. Wieder, "Trace reconstruction with constant deletion probability and related results," in *Proc. ACM-SIAM Symp. Discrete Algorithms*, pp. 389-398, 2008.

[3] S. Kannan and A. McGregor,"More on reconstructing strings from random traces: insertions and deletions," in *Proc. IEEE Intl. Symp. Info. Theory*, Adelaide, Australia, June 2005.

[4] K. Viswanathan and R. Swaminathan, "Improved string reconstruction over insertion-deletion channels," in *Proc. ACM-SIAM Symp. Discrete Algorithms*, San Francisco, CA, 2008.

[5] G. Benson and L. Dong, "Reconstructing the duplication history of a tandem repeat," *ISMB,* 1999, pp. 44-53.

[6] F. Farnoud, M. Schwartz, and J. Bruck, "Estimating mutation rates and sequence age under a stochastic model for tandem duplication and point mutation," available, https://dl.dropboxusercontent.com/u/2041685/website_docs/papers/2015--Estimating%20Mutation%20Rates%20and%20Sequence%20Age.pdf.

[7] S. Jain and F. Farnoud and M. Schwartz, and J. Bruck, "Duplication-correcting codes for data storage in the DNA of living organisms," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Barcelona, July 2016.

[8] F. Farnoud, M. Schwartz, and J. Bruck, "The capacity of string-duplication systems," in *IEEE Trans. Info. Theory*, vol. 62, no. 2, pp. 811-824, Feb. 2016.

[9] R. Gabrys, E. Yaakobi, and O. Milenkovic, "Codes in the Damerau distance for DNA storage," available: http://arxiv.org/abs/1601.06885.

[10] S. M. Yazdi, H. M. Kiah, E. R. Garcia, J. Ma, H. Zhao, and O. Milenkovic, "DNA-based storage: trends and methods," available: http://arxiv.org/abs/1507.01611.

[11] R. Gabrys, H. M. Kiah, and O. Milenkovic, "Asymmetric Lee distance codes for DNA-based storage," *proc. IEEE Int. Symp. Information Theory (ISIT)*, pp. 909-913, Hong Kong, June 2015.

[12] H. M. Kiah, G. J. Puleo, and O. Milenkovic, "Codes for DNA storage channels," *proc. IEEE Information Theory Workshop (ITW)*, pp. 1-5, 2015.

[13] S. M. Yazdi, Y. Yuan, J. Ma, H. Zhao and O. Milenkovic, "A rewritable, random-access DNA-based storage system," available: http://arxiv.org/abs/1505.02199.

[14] R. N. Grass, R. Heckel, M. Puddu, D. Paunescu, and W. J. Stark, "Robust chemical preservation of digital information on DNA in silica with error-correcting codes," *Angewandte Chemie International Edition*, vol. 54, no. 8, Feb. 2015, pp. 2552-2555.

[15] J. Bornholt, R. Lopez, D. M. Carmean, L. Ceze, G. Seelig, and K. Strauss, "A DNA-based archival storage system," to appear in proc. *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, April 2016.

[16] J. Acharya, H. Das, O. Milenkovic, A. Orlitsky, and S. Pan, "String reconstruction from substring compositions," 2014. Available: http://arxiv.org/pdf/1403.2439v1.pdf.

[17] H. M. Kiah, G. J. Puleo, and O. Milenkovic, "Codes for DNA sequence profiles," 2015. Available: http://arxiv.org/pdf/1502.00517v1.pdf

[18] Y. Liron and M. Langberg, "A characterization of the number of subsequences obtained via the deletion channel," *IEEE Trans. Info. Theory*, vol. 61, no. 5, pp. 2300-2312, May 2015.

[19] A. A. Kulkarni and N. Kiyavash, "Non-asymptotic upper bounds for single-deletion correcting codes," *IEEE Trans. Info. Theory*, vol. 59, no. 8, pp. 5115-5130, 2013.

[20] D. Cullina, N. Kiyavash, and A. A. Kulkarni, "Restricted composition deletion correcting codes," to appear in *IEEE Trans. Info. Theory*, 2016.
[21] I. Shomorony, T. Courtade, and D. Tse, "Do read errors matter for genome assembly?" Available: http://arxiv.org/abs/1501.06194.
[22] E. Yaakobi and J. Bruck, "On the uncertainty of information retrieval in associative memories," *Proc. IEEE Intl. Symp. Info. Theory*, Cambridge, MA, July 2012.
[23] V. Junnila and T. Laihonen, "Codes for information retrieval with small uncertainty," *IEEE Trans. Info. Theory*, vol. 60, no. 2, pp. 976-985, Feb. 2014.
[24] V. Junnila and T. Laihonen, "Information retrieval with varying number of input clues," *IEEE Trans. Info. Theory*, vol. 62, no. 2, pp. 625-638, Feb. 2016.
[25] V.I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet Physics Doklady*, vol. 10, no. 8, 1966.
[26] V.I. Levenshtein, "Efficient reconstruction of sequences from their subsequences or supersequences," *Journal of Combinatorial Theory*, vol. 93, no. 2, pp. 310-332, 2001.
[27] V.I. Levenshtein, "Efficient reconstruction of sequences," *Trans. Info. Theory,* vol. 47, no. 1, pp. 2-22, Jan. 2001.
[28] H. S. Wilf, *Generatingfunctionology*. San Diego, CA: Academic Press, 1990.
[29] R. R. Varshamov and G. M. Tenengolts, "Codes which correct single asymmetric errors", *Avtom. i Telemehk.*, vol. 26, no. 2, pp. 288-292, 1965.
[30] F. Sala, C. Schoeny, R. Gabrys, and L. Dolecek, "Three novel combinatorial theorems for the insertion/deletion channel" in *Proc. IEEE Intl. Symp. Info. Theory*, Hong Kong, June 2015.
[31] F. Sala, R. Gabrys, C. Schoeny, K. Mazooji, and L. Dolecek, " Exact sequence reconstruction for insertion-correcting codes" in *Proc. IEEE Intl. Symp. Info. Theory*, Barcelona, July 2016.
[32] R. Gabrys and E. Yaakobi, "Sequence reconstruction over the deletion channel," in *Proc. IEEE Intl. Symp. Info. Theory*, Barcelona, July 2016.
[33] L. Calabi and W.E. Hartnett, "Some general results of coding theory with applications to the study of codes for the correction of synchronization errors," *Information and Control*, vol. 15, no. 3, 1969.
[34] D. Cullina, A. Kulkarni, and N. Kiyavash, "A coloring approach to constructing deletion correcting codes from constant weight subgraphs," in *Proc. IEEE Int. Symp. Info. Theory (ISIT)*, Cambridge, MA, Jul. 2012, pp. 513-517.

## VIII. Appendix

### A. Proof of Lemma 6

This part of the appendix is dedicated to a proof of Lemma 6, restated below.

**Lemma 6.** *For $n$ a positive integer and $t,k,\ell$ non-negative integers such that $t \geqslant k \geqslant \ell$,*

$$\mathcal{N}_2^+(n,t,k,\ell) = \mathcal{N}_2^+(n-1,t,k,\ell) + \mathcal{N}_2^+(n,t-1,k-1,\ell),$$

*and*

$$\mathcal{N}_2^+(n,t,k,\ell) = \mathcal{N}_2^+(n,t-1,k,\ell) + \mathcal{N}_2^+(n-1,t,k-1,\ell-1).$$

*Proof:* We have that

$$\begin{aligned}
&\mathcal{N}_2^+(n-1,t,k,\ell) + \mathcal{N}_2^+(n,t-1,k-1,\ell) \\
&= \sum_{j=\ell}^{k} \binom{t-k+2j}{j}\binom{n-1+k-(2j+1)}{k-j} + \sum_{j=\ell}^{k-1} \binom{(t-1)-(k-1)+2j}{j}\binom{n+(k-1)-(2j+1)}{(k-1)-j} \\
&= \sum_{j=\ell}^{k} \binom{t-k+2j}{j}\binom{n-1+k-(2j+1)}{k-j} + \sum_{j=\ell}^{k-1} \binom{t-k+2j}{j}\binom{n-1+k-(2j+1)}{k-1-j} \\
&= \sum_{j=\ell}^{k-1} \binom{t-k+2j}{j}\left[\binom{n-1+k-(2j+1)}{k-j} + \binom{n-1+k-(2j+1)}{k-1-j}\right] + \binom{t+k}{k} \\
&= \sum_{j=\ell}^{k-1} \binom{t-k+2j}{j}\binom{n+k-(2j+1)}{k-j} + \binom{t+k}{k} \\
&= \sum_{j=\ell}^{k} \binom{t-k+2j}{j}\binom{n+k-(2j+1)}{k-j},
\end{aligned}$$

which is just $\mathcal{N}_2^+(n,t,k,\ell)$, as desired. Next,

$$\begin{aligned}
&\mathcal{N}_2^+(n,t-1,k,\ell) + \mathcal{N}_2^+(n-1,t,k-1,\ell-1) \\
&= \sum_{j=\ell}^{k} \binom{(t-1)-k+2j}{j}\binom{n+k-(2j+1)}{k-j} + \sum_{j=\ell-1}^{k-1} \binom{t-(k-1)+2j}{j}\binom{n-1+(k-1)-(2j+1)}{k-1-j} \\
&= \sum_{j=\ell}^{k} \binom{t-1-k+2j}{j}\binom{n+k-(2j+1)}{k-j} + \sum_{j=\ell-1}^{k-1} \binom{t-k+1+2j}{j}\binom{n+k-2-(2j+1)}{(k-1)-j} \\
&= \sum_{j=\ell}^{k} \binom{t-1-k+2j}{j}\binom{n+k-(2j+1)}{k-j} + \sum_{j=\ell}^{k} \binom{t-k-1+2j}{j-1}\binom{n+k-(2j+1)}{k-j}
\end{aligned}$$

$$= \sum_{j=\ell}^{k} \binom{n+k-(2j+1)}{k-j} \left[ \binom{t-1-k+2j}{j} + \binom{t-k-1+2j}{j-1} \right]$$
$$= \sum_{j=\ell}^{k} \binom{n+k-(2j+1)}{k-j} \binom{t-k+2j}{j},$$

which is indeed $\mathcal{N}_2^+(n,t,k,\ell)$. Here, in the third step, we changed the range of summation for the second term from $[\ell-1, k-1]$ to $[\ell, k]$. ■

### B. *Proof of Lemma* 9

Next, we present the proof of the two auxiliary combinatorial identities.

**Lemma 9.**

1. *For* $m \geqslant 0$,
$$\sum_{j=0}^{m} \binom{2j}{j} \binom{m+j}{2j} (-1)^{m+j} = 1.$$

2. *For* $n, m, t, j \geqslant 0$ *and* $t + j \geqslant m$,
$$\sum_{i=0}^{m} \binom{t+j-i}{t+j-m} \binom{n+t}{i} (-1)^{m-i} = \binom{n+m-j-1}{m}.$$

*Proof:* Both identities will be proved by a generating function approach. This strategy is described as the "snake oil method" in [28]. The idea is that the right-hand side of each identity has an easily-derived generating function, while we will perform more complex manipulations to derive an identical generating function for the left-hand side. For the first identity, the generating function $F(x)$ for the left-hand side is written as

$$\begin{aligned}
F(x) &= \sum_{m \geqslant 0} x^m \sum_{j=0}^{m} \binom{2j}{j} \binom{m+j}{2j} (-1)^{m+j} \\
&= \sum_{j=0}^{\infty} \sum_{m \geqslant j} x^m \binom{2j}{j} \binom{m+j}{2j} (-1)^{m+j} \\
&= \sum_{j=0}^{\infty} \binom{2j}{j} x^{-j} \sum_{m \geqslant j} \binom{m+j}{2j} (-x)^{m+j} \\
&= \sum_{j=0}^{\infty} \binom{2j}{j} x^{-j} \sum_{r' \geqslant 0} \binom{r'}{2j} (-x)^{r'} \\
&= \sum_{j=0}^{\infty} \binom{2j}{j} x^{-j} \frac{(-x)^{2j}}{(1+x)^{2j+1}} \\
&= \frac{1}{1+x} \sum_{j=0}^{\infty} \binom{2j}{j} \left( \frac{x}{(1+x)^2} \right)^j \\
&= \frac{1}{1+x} \frac{1}{\sqrt{1 - \frac{4x}{(1+x)^2}}} \\
&= \frac{1}{1+x} \frac{1+x}{1-x} = \frac{1}{1-x}.
\end{aligned}$$

In the fourth step, we replace $m+j$ with $r'$. We can start the sum at $r' = 0$ since the binomial term $\binom{r'}{2j}$ evaluates to 0 for all $m < j$. Next, in the fifth step, we use the series $\sum_{r \geqslant 0} \binom{r}{k} x^r = \frac{x^k}{(1-x)^{k+1}}$ [28]. The only condition for this identity is $2j \geqslant 0$. Next, in the seventh step, we applied the generating function for the central binomial coefficients [28]:

$$\sum_{j \geqslant 0} \binom{2j}{j} x^j = \frac{1}{\sqrt{1-4x}}.$$

Thus we conclude that $F(x) = 1 + x + x^2 + \ldots$, so indeed $\sum_{j=0}^{m} \binom{2j}{j} \binom{m+j}{2j} (-1)^{m+j} = 1$.

We use the same approach for the second identity. The right-hand side of the identity counts the number of ways to distribute $m$ items in $n-j$ buckets. It is easy to see that this quantity has, with respect to $m$, the generating function $(1+x+x^2+\ldots)^{n-j} = (1-x)^{-(n-j)}$. The left-hand side has generating function

$$\begin{aligned}
F(x) &= \sum_{m\geqslant 0} x^m \sum_{i=0}^{m} \binom{t+j-i}{t+j-m}\binom{n+t}{i}(-1)^{m-i} \\
&= \sum_{i=0}^{\infty}\binom{n+t}{i}\sum_{m\geqslant i} x^m \binom{t+j-i}{t+j-m}(-1)^{m-i} \\
&= \sum_{i=0}^{\infty}\binom{n+t}{i}x^i \sum_{r\geqslant 0} x^r \binom{t+j-i}{t+j-(r+i)}(-1)^{r} \\
&= \sum_{i=0}^{\infty}\binom{n+t}{i}x^i \sum_{r\geqslant 0} x^r \binom{t+j-i}{r}(-1)^{r} \\
&= \sum_{i=0}^{\infty}\binom{n+t}{i}x^i (1-x)^{t+j-i} \\
&= (1-x)^{t+j}\sum_{i=0}^{\infty}\binom{n+t}{i}\left(\frac{x}{1-x}\right)^i \\
&= (1-x)^{t+j}\left(1+\frac{x}{1-x}\right)^{n+t} \\
&= (1-x)^{t+j}(1-x)^{-(n+t)} \\
&= (1-x)^{j-n},
\end{aligned}$$

and we are done. In the third step, we write $m-i=r$. In the fifth and seventh steps, we applied the binomial theorem. ■

### C. Proof of Lemma 13

Finally, we present a proof of Lemma 13, which we restate below:

**Lemma 13.** *For $n\geqslant 1, q\geqslant 2$ and $t,k,\ell\geqslant 1$ with $t\geqslant k\geqslant \ell$, $\mathcal{N}_q^+(n,t,k,\ell)$ satisfies the recursions*

$$\mathcal{N}_q^+(n,t,k,\ell) = \mathcal{N}_q^+(n-1,t,k,\ell) + (q-1)\mathcal{N}_q^+(n,t-1,k-1,\ell),$$

*and*

$$\mathcal{N}_q^+(n,t,k,\ell) = \mathcal{N}_q^+(n,t-1,k,\ell) + \mathcal{N}_q^+(n-1,t,k-1,\ell-1) + (q-2)\mathcal{N}_q^+(n,t-1,k-1,\ell).$$

*Proof:* The proofs of these formulas use only standard sum manipulations and binomial identities. We first show the series of equalities and then describe the steps. For the first recursion, we have

$$\begin{aligned}
&\mathcal{N}_q^+(n-1,t,k,\ell) + (q-1)\mathcal{N}_q^+(n,t-1,k-1,\ell) \\
&\quad\overset{(a)}{=} \sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{(n-1)+t}{i}(q-1)^i(-1)^{k+j-i} + (q-1)\times \\
&\qquad\left[\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-1-j}\binom{(t-1)-(k-1)+2j}{j}\binom{(t-1)+j-i}{(t-1)-(k-1)+2j}\binom{n+(t-1)}{i}(q-1)^i(-1)^{k-1+j-i}\right] \\
&\quad\overset{(b)}{=} \sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n-1+t}{i}(q-1)^i(-1)^{k+j-i} \\
&\qquad+\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-1-j}\binom{t-k+2j}{j}\binom{t-1+j-i}{t-k+2j}\binom{n+t-1}{i}(q-1)^{i+1}(-1)^{k-1+j-i} \\
&\quad\overset{(c)}{=} \sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n-1+t}{i}(q-1)^i(-1)^{k+j-i} \\
&\qquad+\sum_{j=\ell}^{k-1}\sum_{i=1}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n+t-1}{i-1}(q-1)^i(-1)^{k+j-i}
\end{aligned}$$

$$
\begin{aligned}
&\overset{(d)}{=} \sum_{j=\ell}^{k}\sum_{i=1}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n-1+t}{i}(q-1)^i(-1)^{k+j-i}\\
&\quad+\sum_{j=\ell}^{k}\binom{t-k+2j}{j}\binom{t+j}{t-k+2j}(-1)^{k+j}\\
&\quad+\sum_{j=\ell}^{k-1}\sum_{i=1}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n+t-1}{i-1}(q-1)^i(-1)^{k+j-i}\\
&\overset{(e)}{=} \sum_{j=\ell}^{k-1}\sum_{i=1}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}(q-1)^i(-1)^{k+j-i}\left[\binom{n-1+t}{i}+\binom{n+t-1}{i-1}\right]\\
&\quad+\sum_{j=\ell}^{k}\binom{t-k+2j}{j}\binom{t+j}{t-k+2j}(-1)^{k+j}\\
&\overset{(f)}{=} \sum_{j=\ell}^{k-1}\sum_{i=1}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n+t}{i}(q-1)^i(-1)^{k+j-i}\\
&\quad+\sum_{j=\ell}^{k}\binom{t-k+2j}{j}\binom{t+j}{t-k+2j}(-1)^{k+j}\\
&\overset{(g)}{=} \sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n+t}{i}(q-1)^i(-1)^{k+j-i}.\\
&=\mathcal{N}_q^+(n,t,k,\ell).
\end{aligned}
$$

In step $(c)$, we changed the range of summation for $i$ from $[0,k-j-1]$ to $[1,k-j]$ for the second term. In $(d)$, we broke up the sum for the first term, removing the components with $i=0$ in the inner sum. In step $(e)$, we note that there is no inner sum for $j=k$, so we change the limit of the outer sum to $k-1$. We then combined terms. In step $(f)$ we applied the identity $\binom{n+t-1}{i}+\binom{n+t-1}{i-1}=\binom{n+t}{i}$. All other steps are immediate rearrangements of terms.

Next, for the second recursion, we have that

$$
\begin{aligned}
&\mathcal{N}_q^+(n,t-1,k,\ell)+\mathcal{N}_q^+(n-1,t,k-1,\ell-1)+(q-2)\mathcal{N}_q^+(n,t-1,k-1,\ell)\\
&\quad\overset{(a)}{=} \sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\binom{(t-1)-k+2j}{j}\binom{(t-1)+j-i}{(t-1)-k+2j}\binom{n+(t-1)}{i}(q-1)^i(-1)^{k+j-i}\\
&\qquad+\sum_{j=\ell-1}^{k-1}\sum_{i=0}^{k-1-j}\binom{t-(k-1)+2j}{j}\binom{t+j-i}{t-(k-1)+2j}\binom{(n-1)+t}{i}(q-1)^i(-1)^{k-1+j-i}+(q-2)\times\\
&\qquad\left[\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-1-j}\binom{(t-1)-(k-1)+2j}{j}\binom{(t-1)+j-i}{(t-1)-(k-1)+2j}\binom{n+t-1}{i}(q-1)^i(-1)^{k-1+j-i}\right]\\
&\quad\overset{(b)}{=} \sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\binom{t-k+2j-1}{j}\binom{t+j-i-1}{t-k+2j-1}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i}\\
&\qquad+\sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\binom{t-k+2j-1}{j-1}\binom{t+j-i-1}{t-k+2j-1}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i}+(q-2)\times\\
&\qquad\left[\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i-1}{t-k+2j}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i-1}\right]\\
&\quad\overset{(c)}{=} \sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\left[\binom{t-k+2j-1}{j}+\binom{t-k+2j-1}{j-1}\right]\binom{t+j-i-1}{t-k+2j-1}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i}+(q-2)\times\\
&\qquad\left[\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i-1}{t-k+2j}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i-1}\right]\\
&\quad\overset{(d)}{=} \sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i-1}{t-k+2j-1}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i}+(q-2)\times
\end{aligned}
$$

$$
\begin{aligned}
&\left[\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i-1}{t-k+2j}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i-1}\right]\\
&\overset{(e)}{=}\sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i-1}{t-k+2j-1}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i}\\
&\quad-\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i-1}{t-k+2j}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i-1}\\
&\quad+\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i-1}{t-k+2j}\binom{n+t-1}{i}(q-1)^{i+1}(-1)^{k+j-i-1}\\
&\overset{(f)}{=}\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-j-1}\binom{t-k+2j}{j}\left[\binom{t+j-i-1}{t-k+2j-1}+\binom{t+j-i-1}{t-k+2j}\right]\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i}\\
&\quad+\sum_{j=\ell}^{k}\binom{t-k+2j}{j}\binom{t+j-(k-j)-1}{t-k+2j-1}\binom{n+t-1}{k-j}(q-1)^{k-j}(-1)^{k+j-(k-j)}\\
&\quad+\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i-1}{t-k+2j}\binom{n+t-1}{i}(q-1)^{i+1}(-1)^{k+j-i-1}\\
&\overset{(g)}{=}\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i}\\
&\quad+\sum_{j=\ell}^{k}\binom{t-k+2j}{j}\binom{n+t-1}{k-j}(q-1)^{k-j}\\
&\quad+\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i-1}{t-k+2j}\binom{n+t-1}{i}(q-1)^{i+1}(-1)^{k+j-i-1}\\
&\overset{(h)}{=}\sum_{j=\ell}^{k-1}\sum_{i=0}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n+t-1}{i}(q-1)^i(-1)^{k+j-i}\\
&\quad+\sum_{j=\ell}^{k}\binom{t-k+2j}{j}\binom{n+t-1}{k-j}(q-1)^{k-j}\\
&\quad+\sum_{j=\ell}^{k-1}\sum_{i=1}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n+t-1}{i-1}(q-1)^i(-1)^{k+j-i}\\
&\overset{(j)}{=}\sum_{j=\ell}^{k-1}\sum_{i=1}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\left[\binom{n+t-1}{i}+\binom{n+t-1}{i-1}\right](q-1)^i(-1)^{k+j-i}\\
&\quad+\sum_{j=\ell}^{k-1}\binom{t-k+2j}{j}\binom{t+j}{t-k+2j}(-1)^{k+j}+\sum_{j=\ell}^{k-1}\binom{t-k+2j}{j}\binom{n+t-1}{k-j-1}(q-1)^{k-j}\\
&\quad+\sum_{j=\ell}^{k}\binom{t-k+2j}{j}\binom{n+t-1}{k-j}(q-1)^{k-j}\\
&\overset{(k)}{=}\sum_{j=\ell}^{k-1}\sum_{i=1}^{k-j-1}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n+t}{i}(q-1)^i(-1)^{k+j-i}\\
&\quad+\sum_{j=\ell}^{k-1}\binom{t-k+2j}{j}\binom{t+j}{t-k+2j}(-1)^{k+j}+\sum_{j=\ell}^{k}\binom{t-k+2j}{j}\binom{n+t}{k-j}(q-1)^{k-j}\\
&\overset{(l)}{=}\sum_{j=\ell}^{k}\sum_{i=0}^{k-j}\binom{t-k+2j}{j}\binom{t+j-i}{t-k+2j}\binom{n+t}{i}(q-1)^i(-1)^{k+j-i}\\
&=\mathcal{N}_q^+(n,t,k,\ell).
\end{aligned}
$$

The steps we used are the following. In $(b)$, we changed the range of summation for $j$ in the middle term from $[\ell-1,k-1]$

to $[\ell, k]$. In $(d)$, we used the identity $\binom{t-k+2j-1}{j} + \binom{t-k+2j-1}{j-1} = \binom{t-k+2j}{j}$. In $(e)$, we broke up the second term from $(d)$, which is multiplied by a factor of $(q-2)$ into two terms, one multiplied by a factor of $(q-1)$ and the other by $(-1)$. In $(f)$, we combined the first two terms from $(e)$. In $(g)$, we used the identity $\binom{t+j-i-1}{t-k+2j-1} + \binom{t+j-i-1}{t-k+2j} = \binom{t+j-i}{t-k+2j}$. In $(h)$, we changed the range of summation for $i$ in the second term from $[0, k-j-1]$ to $[1, k-j]$. In $(j)$, we combined terms and again applied the identity $\binom{n+t-1}{i} + \binom{n+t-1}{i-1} = \binom{n+t}{i}$. We also combined the last two summands, using the identity $\binom{n+t-1}{k-j-1} + \binom{n+t-1}{k-j} = \binom{n+t}{k-j}$. In $(k)$ we combined all remaining terms. ■