Rate-Distance Trade-offs for List-Decodable Insertion-Deletion Codes

Bernhard Haeupler[*]
Carnegie Mellon University & ETH Zurich
haeupler@cs.cmu.edu

Amirbehshad Shahrasbi[*][†]
Microsoft
ashahrasbi@microsoft.com

Abstract

This paper presents general bounds on the highest achievable rate for list-decodable insertion-deletion codes. In particular, we give novel outer and inner bounds for the highest achievable communication rate of any insertion-deletion code that can be list-decoded from any γ fraction of insertions and any δ fraction of deletions. Our bounds simultaneously generalize the known bounds for the previously studied special cases of insertion-only, deletion-only, and zero-rate and correct other bounds that had been reported for the general case.

1   Introduction

Error-correcting codes are classic combinatorial objects that have been extensively studied since late 40s with broad applications in a multitude of communication and storage applications. While error-correcting codes are mostly studied within the setting that concerns symbol substitutions and erasures (i.e., Hamming-type errors), there has been a recent rise of interest in codes that correct from synchronization errors, such as insertions and deletions, from both theoretical [3–9, 11, 14, 16–20, 22, 24, 28] and practical perspectives [1, 2, 10, 12, 31, 33]. Such codes and their relevant qualities are defined in the same fashion as error-correcting codes, except that the minimum distance requirement is with respect to the pairwise *edit distance* between code words.

Compared to error-correcting codes for Hamming errors synchronization codes are far less understood and many fundamental questions about them remain to be explored. One such important question is the rate-distance trade-off for (worst-cases) synchronization errors, i.e., determining the largest rate that any synchronization code can achieve in the presence of a certain amount of synchronization errors. We address this question in the list-decoding setting.

A code is list-decodable if there exists a decoder D which, for any corrupted codeword (within the desired error bounds), outputs a small size list of codewords that is guaranteed to include the uncorrupted codeword. More formally, an insertion-deletion code $C \subseteq \Sigma^n$ (or insdel code, for short) is (γ, δ, L)-list-decodable if there exists a function $D : \Sigma^* \to 2^C$ such that $|D(w)| \leq L$ for every $w \in \Sigma^*$ and for every codeword $x \in C$ and every word w obtained from x by at most $\gamma \cdot n$ insertions and at most $\delta \cdot n$ deletions, it is the case that $x \in D(w)$. The parameter L is called the list-size. These definitions naturally extend to families of codes with increasing block lengths in the usual way: A family of codes is $(\gamma, \delta, L(\cdot))$-list decodable if each member of the family is $(\gamma, \delta, L(n))$-list decodable where n denotes the block length. Often the function L is omitted and a family of q-ary codes $\mathcal{C}$ is said to be (γ, δ)-list decodable if there exist some polynomial function $L(\cdot)$ for which $\mathcal{C}$ is $(\gamma, \delta, L(\cdot))$-list decodable. The rate R of a family of q-ary codes $\mathcal{C}$ is defined as $R = \lim_{n \to \infty} \frac{\log_q |C_n|}{n}$.

The fundamental question studied in this paper is to understand the inherent trade-off between the communication rate of a q-ary list-decodable insdel code and the amount of synchronization errors it can correct, i.e., the error parameters γ and δ. For every fixed alphabet size q, this trade-off can be nicely plotted as a 3D-surface in a 3D-chart which plots the maximum communication rate on the z-axis for all γ and δ (plotted on the x- and y-axes respectively). See Figure 1 for an example of such a 3D-plot.

Of course, determining the exact communication rate values for any q and any non-trivial values of (γ, δ) is beyond the capability of current techniques. Prior work (described in Section 1.2) has furthermore mainly focused on obtaining a better understanding of certain special cases, which correspond to projections or cuts of the general trade-off plot. In particular, Figure 2 shows the 2D cuts/projections onto the xz- and yz-planes, which correspond to the insertion-only setting with $\delta = 0$ and the deletion-only setting with $\gamma = 0$, as well as the projection/cut onto the yz-plane specifying for which error rate combinations of γ and δ the communication rate hits zero. See Figure 2 for examples of these three 2D-projections. It has also been studied how the shape of the 3D-plot changes asymptotically as q gets larger.

1.1   Our Results

This paper is among the first to give results for the entirety of the 3D trade-off between communication rate and the two error rates for every fixed alphabet size q. We primarily focus on giving good outer bounds, i.e., impossibility results proving limits on the best possible communication rate (for
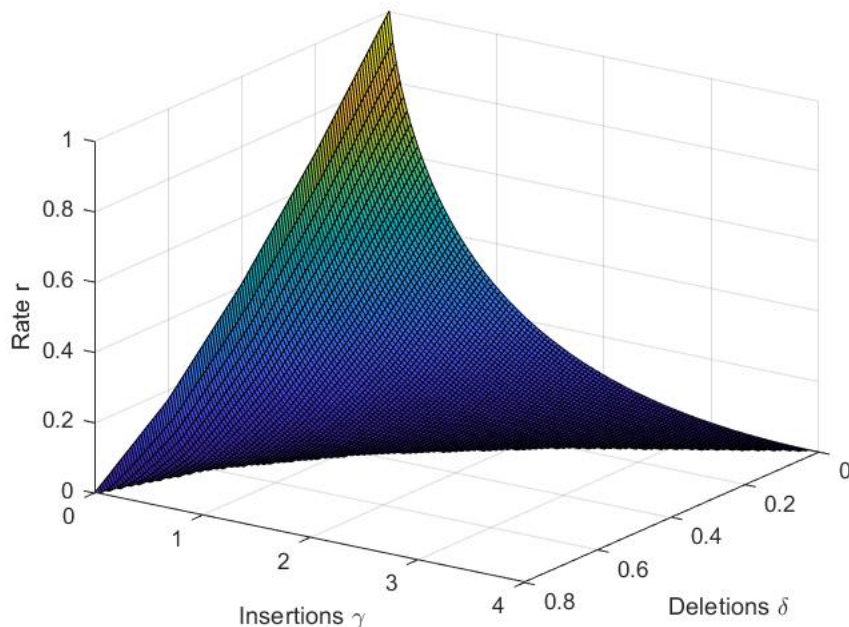
Figure 1: Depiction of our outer bound for $q = 5$.

any given γ, δ, and q). The novel outer bounds we prove are given in the 3D-plot of Figure 2 for an alphabet size of $q = 5$ (similar plots for any given q apply). We develop these outer bounds in Section 2. (See Theorem 2.6)

A notable property of our new outer bound is that, for every q, it exactly matches the best previously known results on all three aforementioned projections/cuts. That is, the outer bound implied for deletion-only codes (i.e., the cut on $\gamma = 0$ plane) matches the deletion-only bound from [23]. Similarly, we match the best insertion-only bounds known (also from [23]) when restricting or projecting our new general outer bound result to the $\delta = 0$ plane. Finally, the error resilience implied by our bound (i.e., where the curve in Figure 2 hits the floor) precisely matches the list-decoding error resilience curve for insertions and deletions as identified by [14]. As such, our bound fully encapsulates and truly generalizes the entirety of the current state-of-the-art of the fundamental rate-distance trade-off for list-decodable insertion-deletion codes for any fixed alphabet size q.

Lastly, for the sake of completeness and as a comparison point, we also provide a general inner bound in Section 3. This general existence result is obtained by analyzing the list-decodability of random codes. We do this mainly through a simple bound on the size of the insertion-deletion sphere. Figure 3 illustrates this inner bound in contrast to the outer bound depicted in Figure 2, also for $q = 5$. It is worth noting that, in contrast to our outer bound, the cut onto the xy-plane does not match the precise error resilience identified (through matching inner and outer bounds) in [14]. However, the cuts onto the xz and yz planes do match the inner bounds of [23] for insertion-only and deletion-only cases, which were also derived by analysis of random codes. We generally believe our outer bounds to be closer to the true zero-error list-decoding channel capacity.
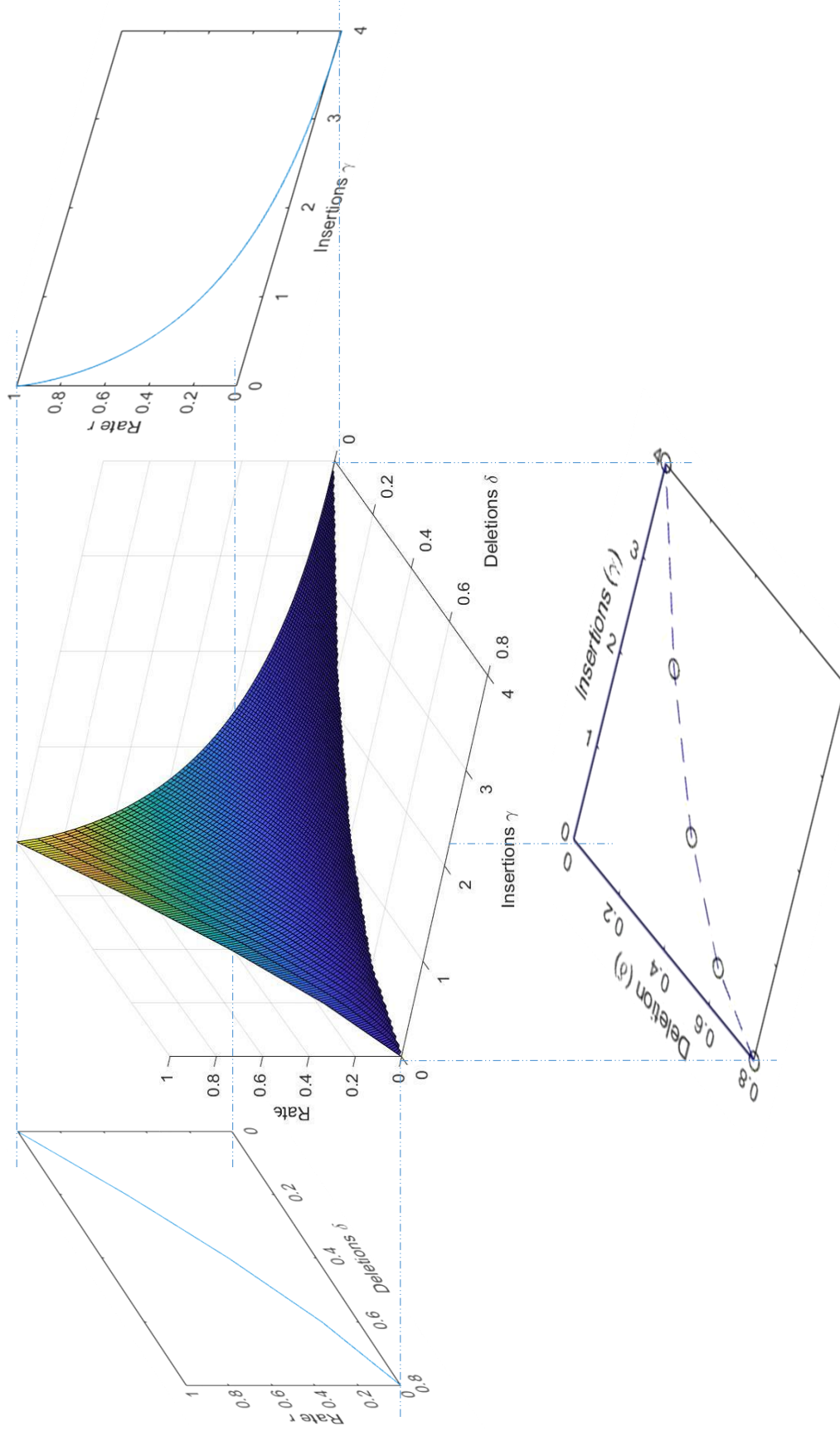
Figure 2: Depiction of our outer bound for $q = 5$ and its three projections for the insertion-only case (on the right), the deletion-only case (on the left), and the zero-rate case (on the bottom). The projection graphs are exactly matching the state-of-the-art results of [14, 23].
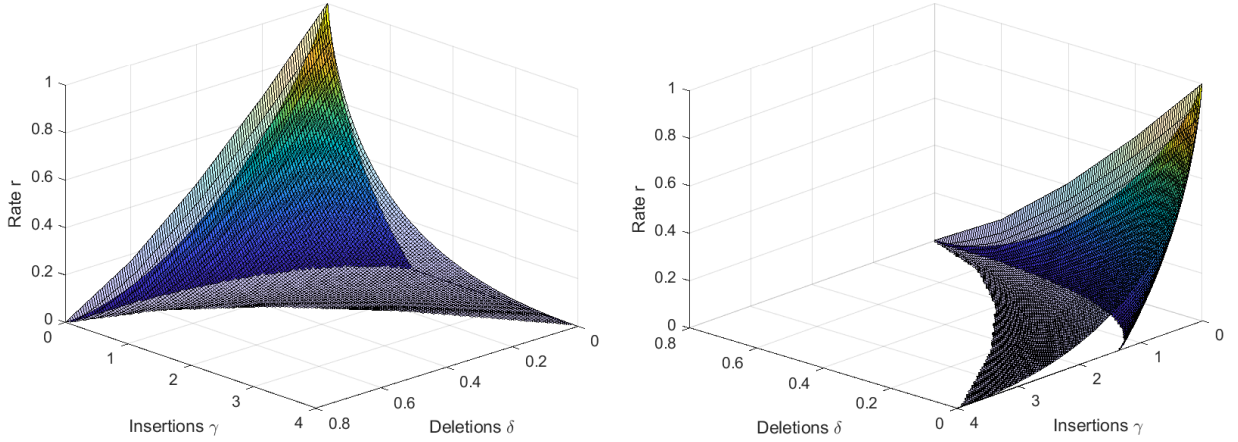
Figure 3: Depiction of our inner and outer bounds for $q = 5$ from two angles. The more transparent surface is the outer bound of Section 2 and the surface underneath is the inner bound derived in Section 3.

1.2 Related Work

This paper studies the fundamental rate-distance trade-off for error correcting codes which are capable of list-decoding from worst-cases insertions and deletions, a topic which has attracted significant attention over the last three years [14, 17, 23, 25, 28, 32]. We summarize these prior works in detail in this section. The multitude of related work on similar questions, such as, (efficient) list-decoding from Hamming errors, unique-decodable insdel codes, or decoding from random insertions or deletions are too many to list or discuss here. Instead, we refer the interested reader to the following (recent) surveys [9, 13, 21, 29, 30], which give detailed accounts of such works.

As noted above, with the exception of [28], mostly special cases of the general rate-distance trade-off for list-decodable insdel codes have been studied up to now. This includes in particular (combinations of) the deletion-only case (with $\gamma = 0$), the insertion-only case (with $\delta = 0$), the zero-rate regime or resilience case asking for what extremal values of (γ, δ) a non-zero rate can be obtained, and the case of large alphabets where the alphabet size $q = O(1)$ is allowed to be a large constant that can depend on the error rates (γ, δ).

1.2.1 List-Decodable Insdel Codes Over Large Constant-Size Alphabets

The rate-distance tradeoff for list-decodable error correcting codes has been studied in [23] under the large alphabet setting, that is the question of finding the largest possible achievable rate that (γ, δ)-list-decodable families of codes can achieve as long as their alphabet size is constant $q = O_{\gamma,\delta}(1)$ (i.e., independent of the block length). Using a method of constructing insdel codes by indexing ordinary error-correcting codes with synchronization strings introduced in [22], [23] shows the following: For every $\delta \in (0, 1)$, $\gamma \geq 0$, and sufficiently small $\varepsilon > 0$, there exists an efficient family of (γ, δ)-list-decodable codes over an alphabet of size $q = O_{\gamma,\delta,\varepsilon}(1)$ that achieve a rate of $1 - \delta - \varepsilon$ or more. It is easy to verify that no such family of codes can achieve a rate larger than $1 - \delta$.

The result of [23] points out an interesting and indeed very drastic distinction between insertions and deletions in the list-decoding setting. In the unique-decoding setting the effect of insertions and deletions are symmetric, and the rate-distance tradeoff can be fully measured solely in terms of the edit-distance between codewords. For list-decoding it turns out that insertions behave completely

4

different than deletions. Indeed while any δ fraction of deletions will definitely reduce the rate at the very least to $1 - \delta$, in the very extreme the impact of insertion errors can be fully compensated by taking the alphabet appropriately large. This is what makes the maximum achievable rate for arbitrarily large constant alphabets merely a function of the deletion error rate δ. This stark distinction in the effects insertions and deletions have on the rate-distance tradeoff for list-decodable codes is the reason why it is crucial to use the two parameters γ and δ to keep track of insertions and deletions separately.

1.2.2 Error Resilience of List-Decodable Insdel Codes

An important special case of the rate-distance trade-off for list-decodable insertion-deletion codes is the question of the best possible *error resilience*. In particular, the question of *"what is the "largest" fraction of errors against which list-decoding is possible for some positive-rate code"* – or differently speaking, at what point(s) the maximum achievable rate of a list-decodable code becomes zero. Understanding this question is, in some way, a prerequisite to meaningfully talk about more general positive rates. Nevertheless, even when restricted to binary deletions-only or insertions-only codes, finding good bounds on the error resilience is highly non-trivial (in contrast to the Hamming case) [14, 15, 17, 25] and has only recently been solved [14]. (along with the general case where insertions and deletions occur together.)

For deletion-only codes, i.e., the special case where $\gamma = 0$, Guruswami and Wang [17] gave binary codes that are list-decodable from a $\delta = \frac{1}{2} - \epsilon$ fraction of errors attaining a $\mathsf{poly}(\epsilon)$ rate for any $\epsilon > 0$. This implies that the error resilience for deletion coding is precisely $\delta_0 = \frac{1}{2}$ since, with a fraction of deletions $\delta \geq \frac{1}{2}$, an adversary can simply eliminate all instances of the least frequent symbol and convert any codeword from $\{0, 1\}^n$ into either $0^{n/2}$ or $1^{n/2}$.

In 2017 a work of Wachter-Zeh [32] gave Johnson-type bounds on list-decodability and list-sizes of codes given their minimum edit-distance. In 2018, Hayashi and Yasunaga [25] made corrections to the results presented in [32], and further showed that such bounds give novel results for the insertion-only case of resilience. In particular, they prove that the codes introduced by Bukh, Guruswami, and Håstad [4] can be list-decoded from up to $\gamma = 0.707$ fraction of insertions (and no deletions) while maintaining a positive-rate.

Very recently, Guruswami et al. [14] improved this fraction of insertions to an optimal $\gamma < 1$. Much more generally [14] were able to tightly and fully identify the error resilience region for codes that are list-decodable from a mixture of insertions and deletions, i.e., determine exactly and for any given q the set of all (γ, δ)s where the largest achievable rate for q-ary (γ, δ)-list decodable codes is non-zero. This fully resolved the zero-rate projection of the question addressed in this paper (shown in the bottom 2D chart of Figure 2).

1.2.3 Alphabet dependent rate results for the deletion-only and insertion-only case

The two other projections, i.e., bounds on the highest achievable rate for the insertion-only ($\delta = 0$) and deletion-only ($\gamma = 0$) cases in dependence on q and the error parameter (γ and δ respectively) were given by Haeupler et al. [23]. These projections are shown in Figure 2 to the right and left respectively. The inner bounds presented in [23] are derived by analyzing list-decoding properties of random codes. Here, we briefly review (the ideas of) the outer bounds from [23] as these will be helpful for the remainder of this paper.

Deletion-only case. A simple observation for deletion-only channels is that no family of positive-rate q-ary codes can be list-decoded from $\delta \geq 1 - \frac{1}{q}$ fraction of deletions. This is due to a simple strategy that adversary can employ to eliminate all occurrences of all symbols of the alphabet except the most frequent one to convert any sent codeword into a word like $a^{n(1-\delta)}$ for some $a \in [q]$. [23] suggests a similar strategy called *Alphabet Reduction* for the adversary when $\delta = \frac{d}{q}$ for some integer d. With $\delta = \frac{d}{q}$ fraction of deletions, an adversary can remove all instances of the d least frequent symbols and, hence, convert any transmitted codeword into a member of an ensemble of $(q-d)^{n(1-\delta)}$ strings. This implies an outer bound of $\frac{\log(q-d)^{n(1-\delta)}}{n \log q} = (1-\delta)\left(1 - \log_q \frac{1}{1-\delta}\right)$ on the largest rate achievable by list-decodable deletion codes for special values of $\delta = \frac{d}{q}$ where $d = 1, 2, \cdots, q-1$. Using a simple time sharing argument between the alphabet reduction strategy over these points, [23] provides a piece-wise linear outer bound for all values of $0 < \delta < 1 - \frac{1}{q}$.

Insertion-only case. In an insertion channel, the received word contains the sent codeword as a subsequence. To provide an outer bound on the highest achievable rate by insertion codes, [23] used the probabilistic method: For a given codeword $x \in [q]^n$, [23] computes the probability of a random string $y \in [q]^{n(1+\gamma)}$ containing x as a subsequence. Having this quantity, one can compute the expected number of codewords of a given code C with rate r that are contained in a random string $y \in [q]^{n(1+\gamma)}$. Note that if r is so high that this expectation is exponentially large in terms of n, then, by linearity of expectation, there exists some string $\bar{y} \in [q]^{n(1+\gamma)}$ which contains exponentially many codewords of C which is a contradiction to its list-decodability from γn insertions. This implies an outer bound for the communication rate which we describe in more details below.

1.2.4 General Case

Liu et al. [27] was the first and only other work studying the rate of list-decodable insertion-deletion codes in full generality, like this paper. After direct contradictions between the results reported here and the claims in [27] were discovered, several correctness issues with key approaches of [27] for outer bounds were identified. These results have been removed in [28], the final version of [27]. The underlying issues seem hard to fix without substantially new ideas, as also reported in the acknowledgements of [28]. As a result, [28] is less directly relevant to this work, with the largest overlap being the inner bounds, similarly derived via a simple analysis of random insertion-deletion codes. Our bound is stronger for all pairs (γ, δ) when $q \geq 3$.

2 Outer Bounds

2.1 Linear Outer Bounds from Resilience Results

We start this section by providing a simple outer bound for best possible rate of an insdel code by generalizing the tight results for the resilience region of [14] into a rate bound for any (γ, δ). Recall that the resilience region F_q for any (integer) alphabet size $q > 2$ is defined as the set of error rates for which there exists list-decodable codes with positive rate, i.e.,

$$F_q = \left\{ (\gamma, \delta) \middle| 0 \leq \delta \leq \frac{q-1}{q}, 0 \leq \gamma \leq q - 1, \exists \ (\gamma, \delta)\text{-list dec. } q\text{-ary code family with positive rate} \right\}.$$

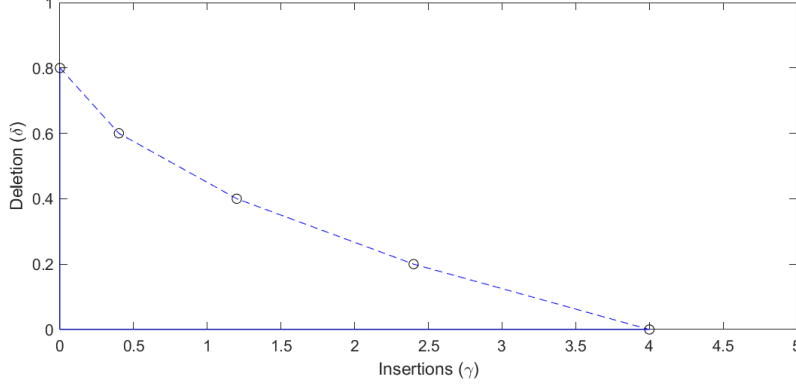In [14], the following exact description of F_q was given:

Figure 4: Resilience region for $q = 5$.

Theorem 2.1 (Theorem 1.3 of [14])**.** *For any positive integer $q \geq 2$, the resilience region F_q is exactly the concave polygon defined over vertices $\left(\frac{i(i-1)}{q}, \frac{q-i}{q} \right)$ for $i = 1, \cdots, q$ and $(0, 0)$, not including the borders except the two segments $[(0, 0), (q - 1, 0))$ and $[(0, 0), (0, 1 - 1/q))$. (See Figure 4) In particular, for any $\varepsilon > 0$ and any $(\gamma, \delta) \in (1 - \varepsilon)F_q$, there exists a family of q-ary codes with positive rate that is (γ, δ)-list-decodable. Further, for any $(\gamma, \delta) \notin F_q$ there exists no (γ, δ)-list-decodable family of q-ary codes with positive rate.*

We show that the multiplicative distance to this resilience region gives a valid outer bound on the rate of any list-decodable insertion-deletion code:

Theorem 2.2. *For any alphabet size q and any $(\gamma, \delta) \in F_q$ let $\alpha \geq 1$ be the smallest number[1] such that $(\alpha\gamma, \alpha\delta) \notin F_q$. Any family of (γ, δ)-list decodable q-ary codes cannot achieve a rate of more than $1 - 1/\alpha$.*

A different way of looking at this outer bound is to think of it as the collection of lines that connect every point on the (border of the) resilience region F_q identified in Theorem 2.1 on the $r = 0$ plane and the point $(\gamma, \delta, r) = (0, 0, 1)$ which indicates the trivial achievable rate of 1 in the absence of noise. (See Figure 5)

The proof of Theorem 2.2 is easy once one recalls how the outer bound for the feasibility region F_q is proven in [14]. It basically consists of a simple strategy transforming any sent string into one of a small $O_q(1)$ number of canonical strings, thus erasing almost all information sent. One can prove Theorem 2.2 by doing the same but only on an $\frac{1}{\alpha}$ fraction of the string. Here though, for the sake of brevity and completeness, we present an alternative and shorter formal proof in the following.

Proof of Theorem 2.2. Assume for the sake of contradiction that for some $(\gamma, \delta) \in F_q$, for which $(\alpha\gamma, \alpha\delta) \notin F_q$, there exists a family of (γ, δ)-list-decodable codes $\mathcal{C}$ with a higher rate than $1 - \frac{1}{\alpha}$, i.e., codes $\mathcal{C} = \{C_1, C_2, \cdots\}$ with block lengths $n_1 < n_2 < \cdots$ and rates $r_1, r_2, \cdots$ that satisfy $r = \lim_{i \to \infty} r_i = 1 - \frac{1}{\alpha} + \varepsilon$ for some $\varepsilon > 0$.

We convert this family of codes to a new family of codes $\mathcal{C}'$ by converting each code C_i into a code C'_i as follows: In all codewords of C_i, consider the $n_i (1 - 1/\alpha)$-long prefix. Among all such prefixes, let p be the most frequent one. We set C'_i to be a code containing all codewords of C_i that start with p. Since all such codewords start with p, we omit the prefix p from all such codewords.

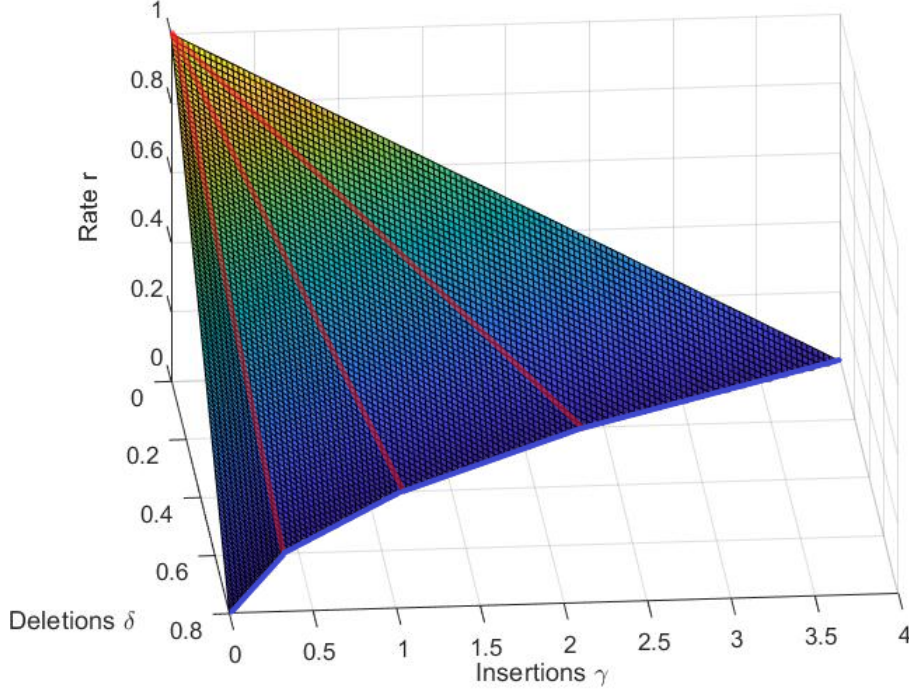[1]such minimum exists due to the definition of F_q.

7

Figure 5: Illustration of the outer bound from Theorem 2.2 for $q = 5$.

Note that the block length of C_i' is $n_i' = n_i - n_i(1 - 1/\alpha) = n_i/\alpha$. Also, since there are $q^{n_i(1-1/\alpha)}$ q-ary strings of length $n_i(1 - 1/\alpha)$,

$$|C_i'| \geq \frac{|C_i|}{q^{n_i(1-1/\alpha)}} = \frac{q^{n_i r_i}}{q^{n_i(1-1/\alpha)}} = q^{n_i(r_i - 1 + 1/\alpha)}.$$

This implies that the rate of C_i' is at least

$$r_i' = \frac{\log_q |C_i'|}{n_i'} \geq \frac{n_i(r_i - 1 + 1/\alpha)}{n_i'} = \alpha(r_i - 1 + 1/\alpha)$$

and, hence, the rate of the family of codes $\mathcal{C}'$ is at least $\lim_{i \to \infty} r_i' \geq \alpha\varepsilon > 0$.

Further, we claim that if $\mathcal{C}$ is $(\gamma, \delta, L(n))$-list decodable, then $\mathcal{C}'$ will be $(\alpha\gamma, \alpha\delta, L(\alpha n'))$-list decodable. To show this, we construct such list-decoder for all codes $C_i' \in \mathcal{C}'$ with input y' by simply padding the most frequent $n_i(1 - 1/\alpha)$-prefix of codewords of $C_i \in \mathcal{C}$, p, in front of y' and running the list-decoder of C_i with input $y = p \cdot y'$. Among the list generated by the decoder of C_i, the ones that do not start with p are withdrawn. The remaining strings will form the output of our list-decoder for C_i' after omitting their prefix p. Note that this indeed gives a $(\alpha\gamma, \alpha\delta, L(\alpha n'))$-list-decoder since for any codeword $x \in C_i'$ that is $(\alpha\gamma, \alpha\delta)$-close to y, $p \cdot x \in C_i$ is (γ, δ)-close to $p \cdot y$.

We were able to show that the family of codes $\mathcal{C}'$ achieves a positive rate and is $(\alpha\gamma, \alpha\delta)$-list decodable for $(\alpha\gamma, \alpha\delta) \notin F_q$. This is a contradiction to Theorem 2.1 proving that the rate of $\mathcal{C}'$ may not exceed $1 - \frac{1}{\alpha}$, thus, proving the theorem. $\qquad\square$

We remark that one can interpret the outer bound from Theorem 2.2 as a convexity argument in the following manner: We know that no code can achieve a rate of one in the presence of

even a small amount of noise. The point $(\gamma, \delta, r) = (0, 0, 1)$ is therefore part of the (in)feasibility boundary. Further, the resilience result from [14] demonstrates that all points within the region $R = \{(\gamma, \delta, 0) \mid (\gamma, \delta) \notin F_q\}$ are infeasible. Theorem 2.2 shows that any convex combination of $(0, 0, 1)$ and any point in R is infeasible as well, implying a pyramid-shaped feasibility region with F_q as its base and $(0, 0, 1)$ as its apex. (See Figure 5.) We remark that generally convex combinations of infeasible points in the rate-distance tradeoff are not known to be infeasible - even for much simpler settings including Hamming errors or unique-decoding. Even convexity results between known infeasible points can be quite challenging. Case in point, our tighter outer bound in Section 2.2 is proven by showing infeasibility of convex combinations of (easier) infeasible points.

2.2 Stronger Bounds Using Generalizations of Bounds from [23]

While Theorem 2.2 gives a bound for all (γ, δ), it is easy to see that it can be quite far from guarantees given by other state-of-the-art outer bounds. This is especially apparent for the deletion-only and insertion-only cases where Theorem 2.2 implies fairly trivial rate bounds of $1 - \frac{\delta}{1 - 1/q}$ and $1 - \frac{\gamma}{q-1}$, respectively. In particular, for the insertion-only case the following outer bound of [23] gives much tighter bounds than $1 - \frac{\gamma}{q-1}$:

Theorem 2.3 (From [23]). *For any alphabet size q and error rate $\gamma < q - 1$, any family of q-ary codes $\mathcal{C}$ which is list-decodable from a γ fraction of insertions has a rate of no more than $1 - \log_q(\gamma + 1) - \gamma \left(\log_q \frac{\gamma + 1}{\gamma} - \log_q \frac{q}{q-1} \right)$.*

Next, we show how to use Theorem 2.3 in a black-box fashion to give a very clean and easily statable outer bound for settings with both insertions and deletions, but in which the fraction of deletions has a nice form, in particular, is a multiple of $\frac{1}{q}$. This outer bound forms the backbone of our final result.

Theorem 2.4. *For any fixed alphabet size q, any insertion rate $\gamma < q - 1$ and any deletion rate $\delta = \frac{d}{q}$ for some integer $d < q$, it is true that any family of q-ary codes $\mathcal{C}$ which is (γ, δ)-list-decodable has a rate of at most $(1 - \delta) \left[\left(1 + \frac{\gamma}{1-\delta} \right) \log_q \frac{q-d}{\frac{\gamma}{1-\delta} + 1} - \frac{\gamma}{1-\delta} \cdot \left(\log_q \frac{q-d-1}{\frac{\gamma}{1-\delta}} \right) \right]$.*

Proof. Consider a code C that is (γ, δ)-list-decodable and assume that $\delta = \frac{d}{q}$ for some integer d. Assume that we restrict the adversary to utilize its deletions in the following manner: The adversary uses the $\frac{d}{q}$ deletion to remove all occurrences of the d-least frequent symbols of the alphabet. If there are remaining deletions, the adversary removes symbols from the end of the transmitted word.

Let us define the code C' that is obtained from C by deleting a δ fraction of symbols from each codeword of C as described above. Note that the block length of C' is $n' = n(1 - \delta)$ and each of its codewords consist of up to $q' = q(1 - \delta) = q - d$ symbols of the alphabet though this subset of size $q - d$ may be different from codeword to codeword. We partition the codewords of C' into $\binom{q}{q-d}$ sets $C'_1, C'_2, \cdots, C'_{\binom{q}{q-d}}$ based on which $(q - d)$-subset of the alphabet they consist of.

Since C is (γ, δ)-list-decodable, each of the C'_is are list-decodable from γn insertions. Therefore, Theorem 2.3 implies that the size of each code C'_i is no larger than

$$q'^{n'}\left[1 - \log_{q'}(\gamma' + 1) - \gamma'\left(\log_{q'} \frac{\gamma' + 1}{\gamma'} - \log_{q'} \frac{q'}{q'-1}\right)\right]$$

where $q' = q - d$, $n' = n(1 - \delta)$, and $\gamma' = \frac{\gamma}{1-\delta}$. Therefore, the size of the code C is no larger than

$$\binom{q}{q-d} q^{n(1-\delta)\left[\log_q q' - \log_q(\gamma' + 1) - \gamma'\left(\log_q \frac{\gamma' + 1}{\gamma'} - \log_q \frac{q'}{q'-1}\right)\right]}$$

9

and, consequently, its rate is no larger than

$$(1-\delta)\left[\log_q(q-d) - \log_q\left(\tfrac{\gamma}{1-\delta}+1\right) - \tfrac{\gamma}{1-\delta}\cdot\left(\log_q\tfrac{\gamma+1-\delta}{\gamma} - \log_q\tfrac{q-d}{q-d-1}\right)\right]$$

$$= (1-\delta)\left[\left(1+\tfrac{\gamma}{1-\delta}\right)\log_q\tfrac{q-d}{\frac{\gamma}{1-\delta}+1} - \tfrac{\gamma}{1-\delta}\cdot\left(\log_q\tfrac{q-d-1}{\frac{\gamma}{1-\delta}}\right)\right].$$

□

Given the nice and explicit form of Theorem 2.4 for any q and γ with multiple specific values of δ, it seems tempting to conjecture that the restriction of δ is unnecessary making $(1-\delta)\left[\left(1+\tfrac{\gamma}{1-\delta}\right)\log_q\tfrac{q-d}{\frac{\gamma}{1-\delta}+1} - \tfrac{\gamma}{1-\delta}\cdot\left(\log_q\tfrac{q-d-1}{\frac{\gamma}{1-\delta}}\right)\right]$ a valid outer bound for any value of δ (and γ). This, however, could not be further from the truth. Indeed, for any δ not of the form restricted to by Theorem 2.4, there exists a γ for which this extended bound is provably wrong because it contradicts the existence of the list-decodable codes constructed in [14].

In fact, for the valid points where δ is a multiple of $\frac{1}{q}$, the rate bound of Theorem 2.4 hits zero at exactly the corner points of the piece-wise linear resilience region F_q characterized by Theorem 2.1. Taking this as an inspiration, one could try to extend the bound of Theorem 2.4 to all values of δ by considering for each q and each rate r the roughly $\frac{q}{r}$ points where Theorem 2.4 hits the plane corresponding to rate r and extend these points in a piece-wise linear manner to a complete 2D-curve for this rate r. This would give a rate bound for any γ, δ, and q as desired, which reduces to a piece-wise linear function for any fixed r and also correctly reproduce F_q for $r = 0$.

It turns out that this is indeed a correct outer bound. However, a stronger form of convexity, which takes full 3D-convex interpolations between any points supplied by Theorem 2.4 and in particular combines points with different rates, also holds and is needed to give our final outer bound.

Theorem 2.5. *For a fixed q, suppose that $(\gamma_0, \delta_0 = \tfrac{d_0}{q})$ and $(\gamma_1, \delta_1 = \tfrac{d_1}{q})$ are two error rate combinations for which Theorem 2.4 implies a maximal communication rate of r_0 and r_1, respectively. For any $0 \le \alpha \le 1$ consider the following convex combinations of these quantities: $\gamma = \alpha\gamma_0 + (1-\alpha)\gamma_1$, $\delta = \alpha\delta_0 + (1-\alpha)\delta_1$, and $r = \alpha r_0 + (1-\alpha)r_1$. It is true that any (δ, γ)-list-decodable q-ary code C has a rate of at most r.*

See Figure 6 for an illustration of this bound for $q = 5$. Red curves indicate the outer bound described above for the special values of δ of the form $\frac{d}{q}$ as given by Theorem 2.4.

Theorem 2.5 together with Theorem 2.4 gives a conceptually very clean description of our outer bound. However, an (exact) evaluation of the outer bound as given by Theorem 2.5 is not straightforward since there are many convex combinations which all produce valid bound but how to compute or select the one which gives the strongest guarantee on the rate for a given (γ, δ) pair is not clear. This is particularly true since, as already mentioned above, the optimal points to combine do not lie on the same rate-plane. To remedy this, we give, as an alternative statement to Theorem 2.5, the next theorem which produces an explicit outer bound for any (γ, δ) as an α-convex combination of two points (γ_0, δ_0) and (γ_1, δ_1) only in dependence on the free parameter γ_0. We then show in Theorem 2.7 an explicit expression for the optimal value for γ_0. Together, this produces a significantly less clean but on the other hand fully explicit description of our outer bound.

Theorem 2.6. *Let C be a q-ary insertion-deletion code that is list-decodable from $\gamma \in [0, q-1]$ fraction of insertions and $\delta \in [0, 1-\tfrac{1}{q}]$ fraction of deletions. Then, the rate of C is no larger than*

$$\alpha\left(1-\frac{d}{q}\right)\left((1+\gamma_0)\log_q\frac{q-d}{1+\gamma_0} - \gamma_0\log_q\frac{q-d-1}{\gamma_0}\right)$$
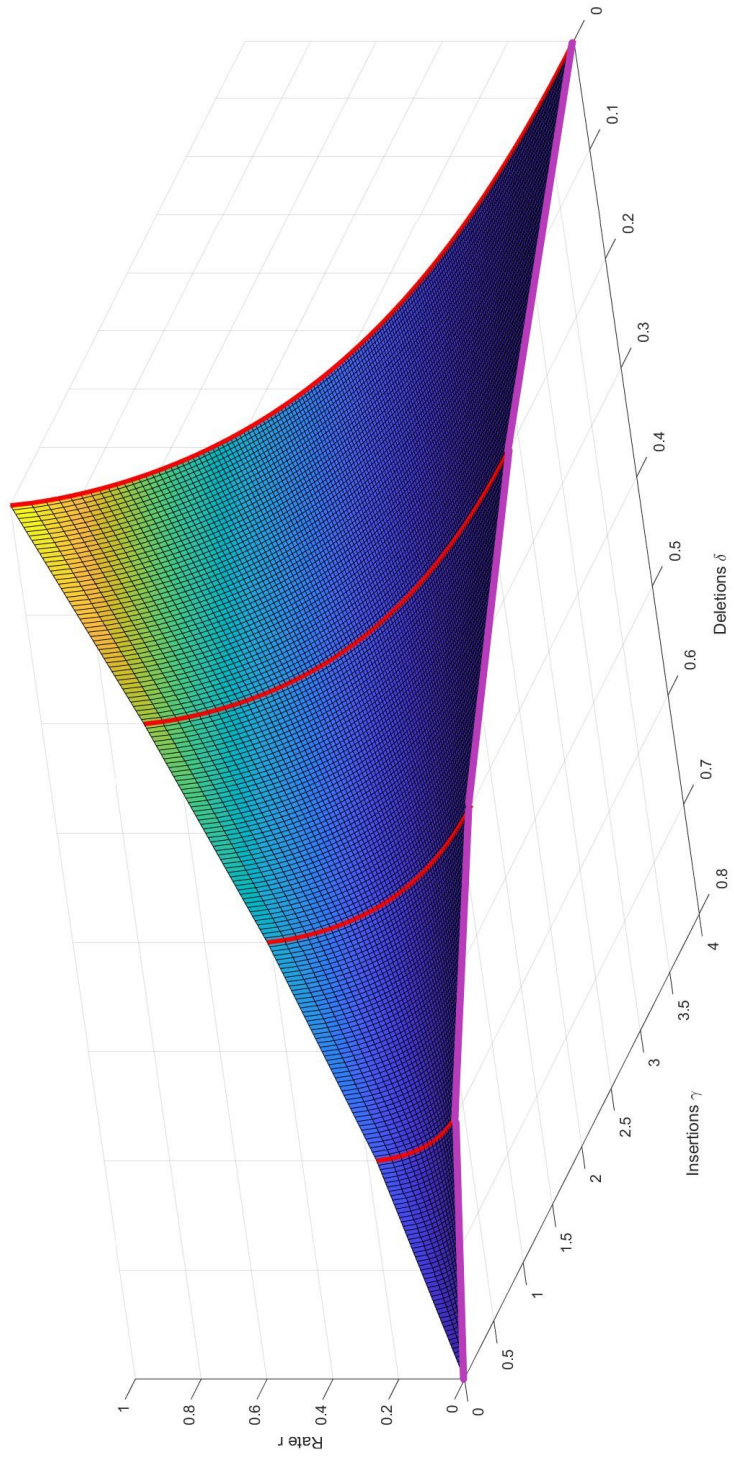
10

Figure 6: Outer bound for rate for $q = 5$. The special case where $\delta = \frac{d}{q}$ for some integer d is indicated with red lines.

$$+ \quad (1-\alpha)\left(1 - \frac{d+1}{q}\right)\left((1+\gamma_1)\log_q \frac{q-d-1}{1+\gamma_1} - \gamma_1 \log_q \frac{q-d-2}{\gamma_1}\right)$$

for $d = \lfloor \delta q \rfloor$, $\alpha = 1 - \delta q + d$, and all $\gamma_0, \gamma_1 \geq 0$ where $\alpha(1 - \frac{d}{q})\gamma_0 + (1-\alpha)(1 - \frac{d+1}{q})\gamma_1 = \gamma$. We present the optimal choice of γ_0 in Theorem 2.7.

Proof of Theorems 2.5 and 2.6. We first note that the statements of Theorem 2.6 and Theorem 2.5 are merely a rephrasing of each other with the exception that Theorem 2.6 only allows and optimizes over convex combinations of neighboring spokes of Theorem 2.4, namely the ones for $d_0 = d$ and $d_1 = d + 1$ for $d = \lfloor \delta q \rfloor$. This restriction, however, is without loss of generality. Indeed, for any values from the domain $\left\{(\gamma, \delta) \mid \delta = \frac{d}{q}, 0 \leq d \leq q-1, d \in \mathbb{Z}\right\}$, Theorem 2.4 gives values which come from the function $f(\gamma, \delta) = (1-\delta)\left[\left(1 + \frac{\gamma}{1-\delta}\right)\log_q \frac{q(1-\delta)}{\frac{\gamma}{1-\delta}+1} - \frac{\gamma}{1-\delta} \cdot \left(\log_q \frac{q(1-\delta)-1}{\frac{\gamma}{1-\delta}}\right)\right]$. This function is convex. (see Appendix A for a formal proof.) Any value given as a convex combination between two non-neighboring spokes can therefore be at least matched (and indeed, thanks to the strict convexity of $f(\cdot, \cdot)$ always be improved) by choosing a different convex combination between neighboring spokes. This justifies the "restricted" formulation of Theorem 2.6, which helps in reducing the number of parameters and simplifies calculations.

In order to prove Theorem 2.6 we, again, fix a specific strategy for the adversary's use of deletions. In particular, the adversary will use $n\alpha\frac{d}{q}$ deletions on the first $n\alpha$ symbols of the transmitted codeword to eliminate all instances of the d least-frequent symbols there. Similarly, he removes all instances of the respective $d+1$ least frequent symbols from the last $n(1-\alpha)$ symbols of the codeword. The resulting string is one out of some $\Sigma_0^{n\alpha(1-\frac{d}{q})} \times \Sigma_1^{n(1-\alpha)(1-\frac{d+1}{q})}$ where $\Sigma_0, \Sigma_1 \subseteq [q]$, $q_0 = |\Sigma_0| = q - d$, $q_1 = |\Sigma_1| = q - d - 1$. This deletion strategy fits within the budgeted number of deletions since $\delta = \alpha\frac{d}{q} + (1-\alpha)\frac{d+1}{q}$ for $d = \lfloor \delta q \rfloor$ and $\alpha = 1 - \delta q + d$.

Note that while the adversary can convert any codeword of C to a string of such form, the sub-alphabets Σ_0 and Σ_1 will likely be different between different codewords of C. Let (Σ_0, Σ_1) be the pair of the most frequently reduced to alphabets and let C_0 be the set of codewords of C that, after undergoing the above-described procedure, turn into a string out of $\Sigma_0^{n\alpha(1-\frac{d}{q})} \times \Sigma_1^{n(1-\alpha)(1-\frac{d+1}{q})}$. Note that $\frac{|C|}{\binom{q}{d}\binom{q}{d+1}} \leq |C_0| \leq |C|$. Further, let D_0 be the set of codewords in C_0 after undergoing the alphabet reduction procedure mentioned above. To give an outer bound of the rate of C it thus suffices to bound from above the size of C_0–or equivalently, D_0; Since C is ($L = \mathsf{poly}(n)$)-list-decodable, no more than L members of C_0 can be mapped to a single member of D_0 during the above procedure, thus, $|D_0| \geq \frac{|C_0|}{\mathsf{poly}(n)}$ which results in a negligible $o(1)$ difference in the rate.

We bound above the size of D_0 by showing that if $|D_0|$ is too large, there will be some received word that can be obtained by exponentially many words in D_0 after $n\gamma$ insertions. Similar to [23], we utilize the linearity of expectation to derive this. Let us pick a random string $Z = (Z_0, Z_1)$ that consists of $n\alpha(1 - \frac{d}{q})(1 + \gamma_0)$ symbols chosen uniformly out of Σ_0 (referred to by Z_0) and $n(1-\alpha)(1 - \frac{d+1}{q})(1 + \gamma_1)$ symbols uniformly chosen out of Σ_1 (referred to by Z_1). We have that $\alpha(1 - \frac{d}{q})\gamma_0 + (1-\alpha)(1 - \frac{d+1}{q})\gamma_1 = \gamma$. ($\gamma_0$ and γ_1 will be determined later.) We calculate the expected number of the members of D_0 that are subsequences of such string – denoted by X. In the following, we will often describe members of D_0 like y as the concatenation (y_0, y_1) where

12

$|y_0| = n_0 = n\alpha(1 - \frac{d}{q})$ and $|y_1| = n_1 = n(1 - \alpha)(1 - \frac{d+1}{q})$.

$$
\begin{aligned}
\mathbb{E}[X] &= \sum_{y=(y_0,y_1)\in D_0} \Pr\{y \text{ is a subsequence of } Z\} \\
&\geq \sum_{y=(y_0,y_1)\in D_0} \Pr\{y_0 \text{ is a subsequence of } Z_0\} \cdot \Pr\{y_1 \text{ is a subsequence of } Z_1\} \\
&= \sum_{y=(y_0,y_1)\in D_0} \prod_{j=0,1} \Pr\{y_j \text{ is a subsequence of } Z_j\} \\
&= \sum_{y=(y_0,y_1)\in D_0} \prod_{j=0,1} \sum_{1\leq a_1<a_2<\cdots<a_{n_j}\leq n_j(1+\gamma_j)} \frac{1}{|\Sigma_j|^{n_j}} \left(1 - \frac{1}{|\Sigma_j|}\right)^{a_{n_j}-n_j} \quad (1) \\
&= |D_0| \prod_{j=0,1} (|\Sigma_j| - 1)^{-n_j} \sum_{l=n_j}^{n_j(1+\gamma_j)} \binom{l-1}{n_j-1} \left(\frac{|\Sigma_j|-1}{|\Sigma_j|}\right)^l \\
&\geq |D_0| \prod_{j=0,1} (|\Sigma_j| - 1)^{-n_j} \binom{n_j(1+\gamma_j)-1}{n_j-1} \left(\frac{|\Sigma_j|-1}{|\Sigma_j|}\right)^{n_j(1+\gamma_j)} \quad (2) \\
&= |D_0| \prod_{j=0,1} (|\Sigma_j| - 1)^{-n_j} \frac{1}{1+\gamma_j} \binom{n_j(1+\gamma_j)}{n_j} \left(\frac{|\Sigma_j|-1}{|\Sigma_j|}\right)^{n_j(1+\gamma_j)} \\
&= |D_0| \prod_{j=0,1} (|\Sigma_j| - 1)^{n_j\gamma_j} |\Sigma_j|^{-n_j(1+\gamma_j)} 2^{n_j(1+\gamma_j)H\left(\frac{1}{1+\gamma_j}\right)+o(n)} \\
&= |D_0| \prod_{j=0,1} q^{n_j\left(\gamma_j \log_q(q_j-1)-(1+\gamma_j)\log_q q_j+(1+\gamma_j)\log_q(1+\gamma_j)-\gamma_j\log_q\gamma_j+o(1)\right)} \\
&= |D_0| \prod_{j=0,1} q^{n_j\left(\gamma_j \log_q \frac{q_j-1}{\gamma_j}-(1+\gamma_j)\log_q \frac{q_j}{1+\gamma_j}+o(1)\right)} \\
&= |D_0| q^{\sum_{j=0,1} n_j\left(\gamma_j \log_q \frac{q_j-1}{\gamma_j}-(1+\gamma_j)\log_q \frac{q_j}{1+\gamma_j}+o(1)\right)} \quad (3)
\end{aligned}
$$

Step (1) is obtained by conditioning the probability of y_j being a subsequence of Z_j over the leftmost occurrence of y_j in Z_j indicated by $a_1, a_2, \cdots, a_n$ as indices of Z_j where the leftmost occurrence of y_j is located. In that event, $Z_j[a_i] = y_j[i]$ and $y_j[i]$ cannot appear in $Z_j[a_{i-1}+1, a_i-1]$. Therefore, the probability of this event is $\left(\frac{1}{q_j}\right)^{n_j} \left(1 - \frac{1}{q_j}\right)^{a_n-n_j}$. To verify Step (2), note that we substituted a summation of positive values with one single term among them.

Finally, by (3), there exists some realization of Z to which at least

$$
|D_0| q^{\sum_{j=0,1} n_j\left(\gamma_j \log_q \frac{q_j-1}{\gamma_j}-(1+\gamma_j)\log_q \frac{q_j}{1+\gamma_j}+o(1)\right)}
$$

codewords of $\mathcal{C}$ are subsequences. In order for C to be list-decodable, this quantity needs to be

sub-exponential. Therefore,

$$
\begin{aligned}
r_C \quad &= \quad \frac{\log_q |D_0| + O(1)}{n} \leq \sum_{j=0,1} \frac{n_j}{n} \left((1+\gamma_j) \log_q \frac{q_j}{1+\gamma_j} - \gamma_j \log_q \frac{q_j - 1}{\gamma_j} \right) \\
&= \quad \alpha \left(1 - \frac{d}{q} \right) \left((1+\gamma_0) \log_q \frac{q_0}{1+\gamma_0} - \gamma_0 \log_q \frac{q_0 - 1}{\gamma_0} \right) \\
&\quad + (1-\alpha) \left(1 - \frac{d+1}{q} \right) \left((1+\gamma_1) \log_q \frac{q_1}{1+\gamma_1} - \gamma_1 \log_q \frac{q_1 - 1}{\gamma_1} \right) \\
&= \quad \alpha \left(1 - \frac{d}{q} \right) \left((1+\gamma_0) \log_q \frac{q-d}{1+\gamma_0} - \gamma_0 \log_q \frac{q-d-1}{\gamma_0} \right) \\
&\quad + (1-\alpha) \left(1 - \frac{d+1}{q} \right) \left((1+\gamma_1) \log_q \frac{q-d-1}{1+\gamma_1} - \gamma_1 \log_q \frac{q-d-2}{\gamma_1} \right) \quad (4)
\end{aligned}
$$

Note that (4) is an outer bound for the rate for all choices of $\gamma_0, \gamma_1 \geq 0$ where $\alpha(1 - \frac{d}{q})\gamma_0 + (1 - \alpha)(1 - \frac{d+1}{q})\gamma_1 = \gamma$. $\qquad \square$

Theorem 2.7. *The optimal choice for the value of γ_0[2] in Theorem 2.6 (i.e., the one that yields the tightest bound) satisfies*

$$
\left(1 + \frac{1}{\gamma_1} \right) \left(1 - \frac{1}{q-d-1} \right) = \left(1 + \frac{1}{\gamma_0} \right) \left(1 - \frac{1}{q-d} \right).
$$

Together with the equation $\alpha(1 - \frac{d}{q})\gamma_0 + (1-\alpha)(1 - \frac{d+1}{q})\gamma_1 = \gamma$, this gives the following explicit expression for γ_0 in terms of q, γ, $d = \lfloor \delta q \rfloor$, and $\alpha = 1 - \delta q + d$:

$$
\gamma_0 = \frac{1}{2\alpha(q-d)} \cdot \left(A - \sqrt{B^2 + C} \right)
$$

for $A = 3\alpha d^2 q + d^2 q - 3\alpha dq^2 - 2dq^2 + 4\alpha dq + 2dq + \alpha q^3 - 2\alpha q^2 + q^3 - 2q^2 + \gamma q + q - \alpha d^3 - 2\alpha d^2$

$B = \alpha d^3 + 2\alpha d^2 - 3\alpha d^2 q - d^2 q + 3\alpha dq^2 + 2dq^2 - 4\alpha dq - 2dq - \alpha q^3 + 2\alpha q^2 - q^3 + 2q^2 - \gamma q - q$

$C = 4(\alpha q - \alpha d)\left(\gamma d^2 q - 2\gamma dq^2 + 2\gamma dq + \gamma q^3 - 2\gamma q^2 \right)$

Proof. To find the optimal value for γ_0, we find the choice of γ_0 that minimizes (1). To this end, we calculate the ratio between the values of (1) when $n\gamma$ insertions are distributed between two parts as $(n_0\gamma_0, n_1\gamma_1)$ and when distributed as $(n_0\gamma_0 + 1, n_1\gamma_1 - 1)$. We show that this ratio monotonically increases as one increases γ_0. Therefore, to find the value of γ_0 for which (1) is minimized, one only needs to find the choice of γ_0 for which this ratio is equal to one.

$$
\frac{\binom{n_0(1+\gamma_0)}{n_0} \left(\frac{|\Sigma_0|-1}{|\Sigma_0|} \right)^{n_0(1+\gamma_0)} \binom{n_1(1+\gamma_1)}{n_1} \left(\frac{|\Sigma_1|-1}{|\Sigma_1|} \right)^{n_1(1+\gamma_1)}}{\binom{n_0(1+\gamma_0)+1}{n_0} \left(\frac{|\Sigma_0|-1}{|\Sigma_0|} \right)^{n_0(1+\gamma_0)+1} \binom{n_1(1+\gamma_1)-1}{n_1} \left(\frac{|\Sigma_1|-1}{|\Sigma_1|} \right)^{n_1(1+\gamma_1)-1}}
$$

---

[2]Note that choosing γ_0 also determines a value for γ_1.

$$
\begin{aligned}
&= \frac{\binom{n_0(1+\gamma_0)}{n_0}\binom{n_1(1+\gamma_1)}{n_1}\binom{|\Sigma_1|-1}{|\Sigma_1|}}{\binom{n_0(1+\gamma_0)+1}{n_0}\binom{|\Sigma_0|-1}{|\Sigma_0|}\binom{n_1(1+\gamma_1)-1}{n_1}} \\[2ex]
&= \frac{\binom{n_0(1+\gamma_0)}{n_0}\binom{n_1(1+\gamma_1)}{n_1}}{\binom{n_0(1+\gamma_0)+1}{n_0}\binom{n_1(1+\gamma_1)-1}{n_1}} \times \frac{|\Sigma_0| \times (|\Sigma_1|-1)}{(|\Sigma_0|-1) \times |\Sigma_1|} \\[2ex]
&= \frac{\frac{n_1(1+\gamma_1)}{n_1(1+\gamma_1)-n_1}}{\frac{n_0(1+\gamma_0)+1}{n_0(1+\gamma_0)+1-n_0}} \times \frac{|\Sigma_0| \times (|\Sigma_1|-1)}{(|\Sigma_0|-1) \times |\Sigma_1|} \\[2ex]
&= \frac{\frac{n_1(1+\gamma_1)}{n_1\gamma_1}}{\frac{n_0(1+\gamma_0)+1}{n_0\gamma_0+1}} \times \frac{|\Sigma_0| \times (|\Sigma_1|-1)}{(|\Sigma_0|-1) \times |\Sigma_1|} \\[2ex]
&= \frac{1+\frac{1}{\gamma_1}}{1+\frac{1}{\gamma_0+1/n_0}} \times \frac{1-\frac{1}{|\Sigma_1|}}{1-\frac{1}{|\Sigma_0|}}
\end{aligned}
$$

Note that as one increases γ_0, the numerator grows and the denominator becomes smaller–meaning that the overall value goes up. Therefore, the optimal choice of γ_0 would be one for which:

$$
\frac{1+\frac{1}{\gamma_1}}{1+\frac{1}{\gamma_0+1/n_0}} \times \frac{1-\frac{1}{|\Sigma_1|}}{1-\frac{1}{|\Sigma_0|}} = 1
$$

$$
\Leftrightarrow \quad \left(1+\frac{1}{\gamma_1}\right)\left(1-\frac{1}{|\Sigma_1|}\right) = \left(1+\frac{1}{\gamma_0+1/n_0}\right)\left(1-\frac{1}{|\Sigma_0|}\right)
$$

Given that families of codes with increasing block lengths n are considered, the term $\frac{1}{n_0} = \frac{1}{n \cdot \alpha(1-d/q)}$ vanishes. Thus, we are looking for a choice of γ_0 that satisfies

$$
\left(1+\frac{1}{\gamma_1}\right)\left(1-\frac{1}{|\Sigma_1|}\right) = \left(1+\frac{1}{\gamma_0}\right)\left(1-\frac{1}{|\Sigma_0|}\right).
$$

Putting this together with the equation $\alpha(1-\frac{d}{q})\gamma_0 + (1-\alpha)(1-\frac{d+1}{q})\gamma_1 = \gamma$ from Theorem 2.6 and solving the resulting system of equations analytically using computer software, the stated equation for the optimal choice of γ_0 is derived. $\square$

3 Inner Bound via Analyzing Random Codes

In this section, we provide an inner bound on the highest rate achievable by list-decodable insertion-deletion codes. We start with a preliminary lemma in the following. Throughout this section, we define $\mathcal{B}_i(S, n_i)$ or the *insertion sphere of radius* n_i as the set of all strings that can be obtained by n_i insertions from S. $\mathcal{B}_d(S, n_d)$ and $\mathcal{B}(S, n_i, n_d)$ are similarly defined as the *deletion sphere of radius* n_d and the *insertion-deletion sphere of insertion radius* n_i *and deletion radius* n_d around S.

Lemma 3.1 (From [26]). *Let* n, n_i, *and* q *be positive integers and* $S \in [q]^n$. *Then, the size of the insertion sphere of radius* n_i *around* S *is*

$$
|\mathcal{B}_i(S, n_i)| = \sum_{i=0}^{n_i} \binom{n+n_i}{i}(q-1)^i.
$$

In the following, we give a simple bound on the size of the insertion-deletion sphere.

Lemma 3.2. *Let $x \in [q]^n$, $\delta \in \left[0, 1 - \frac{1}{q}\right]$ and $\gamma \in [0, (q-1)(1-\delta)]$. The size of the insertion-deletion sphere of insertion-radius γn and deletion-radius δn around x is no larger than*

$$|\mathcal{B}(x, \gamma n, \delta n)| \leq q^{n\left(H_q(\delta) + (1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right) - \delta \log_q(q-1)\right) + o(n)}$$

where $H_q(\cdot)$ denotes the q-ary entropy function (see definition in (8)).

Proof.

$$
\begin{aligned}
|\mathcal{B}(x, \gamma n, \delta n)| \quad &\leq \quad \sum_{x_0 \in \mathcal{B}_d(x, \delta n)} |\mathcal{B}_i(x_0, \gamma n)| \\
&\leq \quad \binom{n}{\delta n} \sum_{i=0}^{\gamma n} \binom{n(1-\delta+\gamma)}{i}(q-1)^i \qquad (5) \\
&\leq \quad n\gamma \binom{n}{n\delta}\binom{n(1-\delta+\gamma)}{n\gamma}(q-1)^{\gamma n} \qquad (6) \\
&= \quad q^{n\left(H_q(\delta) - \delta \log_q(q-1) + (1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right) - \gamma \log_q(q-1) + \gamma \log_q(q-1)\right) + o(n)} \qquad (7) \\
&= \quad q^{n\left(H_q(\delta) + (1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right) - \delta \log_q(q-1)\right) + o(n)}
\end{aligned}
$$

Note that (5) follows from Lemma 3.1 and (6) is true because the term in summation reaches its maximum when $i = n\gamma$. To see this, we test the ratio between the value of the term for two consecutive parameter values i and $i+1$:

$$\frac{\binom{n(1-\delta+\gamma)}{i+1}(q-1)^{i+1}}{\binom{n(1-\delta+\gamma)}{i}(q-1)^i} = \frac{n(1-\delta+\gamma)-i}{i+1}(q-1)$$

Note that $\frac{n(1-\delta+\gamma)-i}{i+1}(q-1) \geq 1 \Leftrightarrow \frac{n(1-\delta+\gamma)-i}{i+1} \geq \frac{1}{q-1} \Leftrightarrow iq + 1 \leq n(1-\delta+\gamma)(q-1)$. This holds for all $i \leq n\gamma$ because:

$$n\gamma q + 1 \leq n(1-\delta+\gamma)(q-1) \Leftrightarrow n\gamma < n(1-\delta)(q-1).$$

Finally, (7) follows from the definition of the q-qry entropy function

$$H_q(x) = x \log_q(q-1) - x \log_q x - (1-x)\log_q(1-x) \qquad (8)$$

and the equation $\binom{n}{np} = q^{n(H_q(p) - p \log_q(q-1)) + o(n)}$. $\qquad \square$

Using the bound on the size of the insertion-deletion radius presented above, we give the following inner bound on the highest achievable rate for (γ, δ)-list-decodable codes derived by analysis of the list-decodability of random codes.

Theorem 3.3. *For any integer $q \geq 2$, $\delta \in \left[0, 1 - \frac{1}{q}\right]$ and $\gamma \in [0, (q-1)(1-\delta)]$, a family of random q-ary codes with rate*

$$0 \leq R < 1 - (1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right) - H_q(\delta) + \gamma \log_q(q-1)$$

is list-decodable from γn insertions and δn deletions with high probability.

We remark that the condition $\gamma \leq (q-1)(1-\delta)$ does not weaken the statement of the theorem since there is no positive-rate family of codes that can list-decode from (γ, δ) fraction of errors where $\gamma \geq (1-\delta)(q-1)$. To see this, similar to the proof of Theorem 2.6, one can think of an adversary that reduces the alphabet to one of size $q - d$ in the first $n\alpha$ symbols of the message and to $q - d - 1$ in the rest of it where $d = \lfloor \delta q \rfloor$ and $\alpha = 1 - \delta q + d$. With

$$\gamma n \geq n(1-\delta)(q-1) \geq n\alpha \left(1 - \frac{d}{q}\right)(q-d-1) + n(1-\alpha)\left(1 - \frac{d+1}{q}\right)(q-d-2)$$

insertions, the adversary can turn any sent message into a string out of an ensemble of $\binom{q}{q-d}\binom{q}{q-d-1}$ strings by turning each message into repetitions of the reduce alphabet members as described in the proof of Theorem 2.6.

Proof of Theorem 3.3. Take the random codeword $X \in [q]^n$ and some string $y \in [q]^{n(1-\delta+\gamma)}$ of length $n' = n(1-\delta+\gamma)$. Using Lemma 3.2, the probability of X being inside the insertion-deletion sphere of deletion-radius $\delta'n' = \gamma n$ and insertion-radius $\gamma'n' = \delta n$ of y is

$$\Pr\{X \in \mathcal{B}(y, \gamma'n', \delta'n')\} \leq \frac{q^{n'\left(H_q(\delta') + (1-\delta'+\gamma')H_q\left(\frac{\gamma'}{1-\delta'+\gamma'}\right) - \delta' \log_q(q-1)\right) + o(n)}}{q^n}$$

$$= \frac{q^{n(1-\delta+\gamma)\left(H_q\left(\frac{\gamma}{1-\delta+\gamma}\right) + \frac{1}{1-\delta+\gamma}H_q(\delta) - \frac{\gamma}{1-\delta+\gamma}\log_q(q-1)\right) + o(n)}}{q^n}$$

$$= q^{n\left(H_q(\delta) + (1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right) - \gamma \log_q(q-1) - 1\right) + o(n)}$$

For the random code C with rate R to not be l-list decodable for some integer l, there has to exists some string $y \in [q]^{n(1-\delta+\gamma)}$ that can be obtained by $l+1$ codewords of C via δn deletions and γn insertions, i.e., codewords that lie in $\mathcal{B}(y, \delta n, \gamma n)$. Using the union bound over all $y \in [q]^{n(1-\delta+\gamma)}$, the probability of the existence of such y is at most.

$$q^{n(1-\delta+\gamma)}\left(q^{nR}\right)^{l+1}\left(q^{n\left(H_q(\delta) + (1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right) - \gamma \log_q(q-1) - 1\right) + o(n)}\right)^{l+1}$$

$$= q^{n(l+1)\left(R + H_q(\delta) + (1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right) - \gamma \log_q(q-1) - 1 + \frac{1-\delta+\gamma}{l+1} + o_l(1)\right)} \quad (9)$$

Note that we used the trivial bound $\binom{q^{nR}}{l+1} \leq \left(q^{nR}\right)^{l+1}$ in the above calculation. Equation (9) implies that as long as

$$R < 1 - (1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right) - H_q(\delta) + \gamma \log_q(q-1),$$

for an appropriately large $l = O_{\gamma,\delta,q}(1)$, the exponent of (9) is negative and, therefore, the family of random codes is (γ, δ)-list-decodable with high probability. $\square$

We next show that this bound is stronger than the one presented by [28] for the case of alphabet size $q \geq 3$.

Theorem 3.4. *The bound presented in Theorem 3.3 is stronger (i.e. larger) than the following that is provided in Lemma 15 of [28] for any δ and γ.*

$$1 - (1-\delta+2\gamma)H_q\left(\frac{\gamma}{1-\delta+2\gamma}\right) - H_q(\delta) + \gamma \log_q(q-1)$$

17

Proof. To prove this, one simply needs to verify that

$$(1 - \delta + \gamma)H_q\left(\frac{\gamma}{1 - \delta + \gamma}\right) \le (1 - \delta + 2\gamma)H_q\left(\frac{\gamma}{1 - \delta + 2\gamma}\right)$$

$$\Leftrightarrow \quad -\gamma \log_q\left(\frac{\gamma}{1 - \delta + \gamma}\right) - (1 - \delta)\log_q\left(\frac{1 - \delta}{1 - \delta + \gamma}\right) + \gamma \log_q(q - 1)$$

$$\le -\gamma \log_q\left(\frac{\gamma}{1 - \delta + 2\gamma}\right) - (1 - \delta + \gamma)\log_q\left(\frac{1 - \delta + \gamma}{1 - \delta + 2\gamma}\right) + \gamma \log_q(q - 1)$$

$$\Leftrightarrow \quad -\gamma \log_q\left(\frac{\gamma}{1 - \delta + \gamma}\right) - (1 - \delta)\log_q\left(\frac{1 - \delta}{1 - \delta + \gamma}\right)$$

$$\le -\gamma \log_q\left(\frac{\gamma}{1 - \delta + 2\gamma}\right) - (1 - \delta + \gamma)\log_q\left(\frac{1 - \delta + \gamma}{1 - \delta + 2\gamma}\right)$$

$$\Leftrightarrow \quad 2\gamma \log_q\left(\frac{1 - \delta + \gamma}{1 - \delta + 2\gamma}\right) \le (1 - \delta)\log_q\left(\frac{(1 - \delta)(1 - \delta + 2\gamma)}{(1 - \delta + \gamma)^2}\right)$$

$$\Leftrightarrow \quad 2(1 - \delta + \gamma)\log_q(1 - \delta + \gamma) \le (1 - \delta)\log_q(1 - \delta) + (1 - \delta + 2\gamma)\log_q(1 - \delta + 2\gamma)$$

The last line of the above sequence of equations is true due to the $f\left(\frac{a+b}{2}\right) \le \frac{f(a)+f(b)}{2}$ inequality that holds for the convex function $f(x) = x\log_q x$ and points $a = 1-\delta$ and $b = 1-\delta+2\gamma$. Therefore, the claim we began with is correct. $\qquad \square$

Appendices

A   Proof of the Convexity of $f(\gamma, \delta)$

In this section, we show that the bivariate function

$$f(\gamma, \delta) = (1 - \delta)\left[\left(1 + \frac{\gamma}{1 - \delta}\right)\log_q \frac{q(1 - \delta)}{\frac{\gamma}{1-\delta} + 1} - \frac{\gamma}{1 - \delta} \cdot \left(\log_q \frac{q(1 - \delta) - 1}{\frac{\gamma}{1-\delta}}\right)\right]$$

is convex. To prove the convexity, our general strategy is to show that the Hessian matrix of f is positive semi-definite. In order to do so, we take the following steps: We first characterize a domain D for $f(\gamma, \delta)$, over which we analyze the convexity. We then calculate the Hessian matrix of the function f, H_f. To show the positive semi-definiteness of H_f, we form its characteristic polynomial and then show that both of its solutions are real and non-negative – meaning that both eigenvalues of H_f are non-negative over the domain D. This would imply that H_f is positive semi-definite and, hence, f is convex over D.

Determining the domain D.   Let us begin with describing the domain D. As stated in Section 1.2, for the purposes of this paper, we only consider the error rates that are within $\delta \in [0, 1-1/q]$ and $\gamma \in [0, q-1]$. Note that for any fixed value $\delta \in [0, 1-1/q)$, $f(\gamma = 0, \delta)$ is positive. We will show that as γ grows, the value of $f(\gamma, \delta)$ continuously drops until it reaches zero at $\gamma = (1-\delta)(q-q\delta-1)$. This suggests that the domain D has to be defined as follows:

$$D = \left\{(\gamma, \delta) \mid 0 \leq \delta \leq 1 - \frac{1}{q}, 0 \leq \gamma \leq (1 - \delta)(q - q\delta - 1)\right\}.$$

To show the claim above, we demonstrate two simple facts: (I) The partial derivative of f with respect to γ is negative within $0 \leq \gamma \leq (1 - \delta)(q - q\delta - 1)$ and, (II) $f(\gamma, \delta) = 0$ for $\gamma = (1 - \delta)(q - q\delta - 1)$.

To see claim (a), note that

$$\frac{\partial f}{\partial \gamma} = \log_q \left(\frac{(1 - \delta)^2 q}{1 - \delta + \gamma}\right) - \log_q \left(\frac{(1 - \delta)(q - q\delta - 1)}{\gamma}\right) = \log_q \left(\frac{q(1 - \delta)\gamma}{(1 - \delta + \gamma)(q - q\delta - 1)}\right)$$

which is non-positive as long as

$$\begin{aligned}
\frac{\partial f}{\partial \gamma} \leq 0 \quad &\Leftrightarrow \quad \frac{q(1 - \delta)\gamma}{(1 - \delta + \gamma)(q - q\delta - 1)} < 1 \\
&\Leftrightarrow \quad q(1 - \delta)\gamma \leq (1 - \delta + \gamma)(q - q\delta - 1) \quad\quad\quad (10)\\
&\Leftrightarrow \quad \gamma \leq (1 - \delta)(q - q\delta - 1)
\end{aligned}$$

Note that (10) is valid since $1 - \delta + \gamma \geq 0$ and $\delta \leq 1 - \frac{1}{q} \Rightarrow q - q\delta - 1 \geq 0$. One can also easily evaluate $f(\gamma, \delta)$ for $\gamma = (1 - \delta)(q - q\delta - 1)$ to confirm claim (b).

Hessian Matrix and Characteristic Polynomial.   We now proceed to calculating the Hessian matrix of f and forming its characteristic polynomial.

$$H_f = \begin{bmatrix} H_{1,1} & H_{1,2} \\ H_{2,1} & H_{2,2} \end{bmatrix} = \begin{bmatrix} \frac{\partial^2 f}{\partial \gamma^2} & \frac{\partial^2 f}{\partial \gamma \partial \delta} \\ \\ \frac{\partial^2 f}{\partial \delta \partial \gamma} & \frac{\partial^2 f}{\partial \delta^2} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{1-\delta}{\gamma(1-\delta+\gamma)\log(q)} & \frac{\gamma+(1-\delta)^2 q}{(1-\delta)(1-\delta+\gamma)(q-q\delta-1)\log(q)} \\ \\ \frac{\gamma+(1-\delta)^2 q}{(1-\delta)(1-\delta+\gamma)(q-q\delta-1)\log(q)} & \frac{(1-\delta)^3 q^2(1-\delta+2\gamma)+(2(1-\delta)q-1)\left(\gamma^2-\gamma(1-\delta)-(1-\delta)^2\right)}{(1-\delta)^2(1-\delta+\gamma)(q-q\delta-1)^2 \log(q)} \end{bmatrix} \quad (11)$$

We prove semi-definiteness by deriving the characteristic polynomial of H_f. The eigenvalues of H_f are the roots of this polynomial.

$$\begin{aligned} \det(H_f - \lambda I) = 0 \quad &\Leftrightarrow \quad \begin{vmatrix} H_{1,1} - \lambda & H_{1,2} \\ H_{2,1} & H_{2,2} - \lambda \end{vmatrix} = 0 \\ &\Leftrightarrow \quad (H_{1,1} - \lambda)(H_{2,2} - \lambda) - H_{1,2}H_{2,1} = 0 \\ &\Leftrightarrow \quad \lambda^2 - (H_{1,1} + H_{2,2})\lambda + (H_{1,1}H_{2,2} - H_{1,2}H_{2,1}) = 0 \quad (12) \end{aligned}$$

To prove the semi-definiteness of H_f, we show that both of its eigenvalues are non-negative, or equivalently, the roots of the quadratic equation (12) are both non-negative. We remind the reader of the straightforward fact that in a quadratic equation of form $x^2 - Sx + P = 0$, S is the sum of the roots and P is their product. Therefore, to show that both roots are non-negative, we only need to show that S and P are both non-negative and that the roots are both real, i.e., $\Delta = S^2 - 4P \geq 0$.

1. $H_{1,1} + H_{2,2} \geq 0$

2. $H_{1,1}H_{2,2} - H_{1,2}H_{2,1} \geq 0$

3. $(H_{1,1} + H_{2,2})^2 - 4(H_{1,1}H_{2,2} - H_{1,2}H_{2,1}) \geq 0$

In the remainder of this section, we prove the three items listed above.

Proof of Item 1. Given (11), we have that

$$\begin{aligned} H_{1,1} + H_{2,2} &= \frac{1}{\log(q)} \cdot \left(\frac{1}{\gamma} + \frac{\gamma q^2}{(q - q\delta - 1)^2} + \frac{\gamma(1-\gamma-\delta)}{(1-\delta)^2(1-\delta+\gamma)} \right) \\ &= \frac{1}{\log(q)} \cdot \left(\frac{1}{\gamma} + \frac{\gamma}{(1-1/q-\delta)^2} + \frac{\gamma}{(1-\delta)(1-\delta+\gamma)} - \frac{\gamma^2}{(1-\delta)^2(1-\delta+\gamma)} \right) (13) \end{aligned}$$

Note that terms $\frac{1}{\gamma}$ and $\frac{\gamma}{(1-\delta)(1-\delta+\gamma)}$ are positive. Therefore, to prove that $H_{1,1}+H_{2,2}$ is non-negative, we show that

$$\frac{\gamma}{(1-1/q-\delta)^2} - \frac{\gamma^2}{(1-\delta)^2(1-\delta+\gamma)} \geq 0. \quad (14)$$

Note that

$$1 - \delta - 1/q < 1 - \delta \Rightarrow \frac{\gamma}{(1-1/q-\delta)^2} \geq \frac{\gamma}{(1-\delta)^2}. \quad (15)$$

Also, since $\delta \leq 1$, we have that $1 - \delta + \gamma \geq \gamma \Rightarrow \frac{\gamma}{1-\delta+\gamma} \leq 1$. Thus, (15) holds even if one multiplies its right-hand side by $\frac{\gamma}{1-\delta+\gamma}$ which gives (14) and, thus, proves Item 1.

Proof of Item 2. Given ([11](#)), we have that

$$H_{1,1}H_{2,2} - H_{1,2}H_{2,1} = \frac{(1-\delta)^2(q-q\delta-1)^2 - \gamma^2}{\gamma(1-\delta)^2(1-\delta+\gamma)(q-q\delta-1)^2\log^2(q)}.$$

Note that all terms in the denominator are positive. The numerator is positive as well since, as mentioned earlier, the domain D is defined only to include points (γ, δ) where $\gamma \leq (1-\delta)(q-q\delta-1)$.

Proof of Item 3. This claim can be simply shown to be true as follows:

$$(H_{1,1} + H_{2,2})^2 - 4(H_{1,1}H_{2,2} - H_{1,2}H_{2,1}) = (H_{1,1} - H_{2,2})^2 + 4H_{1,2}H_{2,1} = (H_{1,1} - H_{2,2})^2 + 4H_{1,2}^2$$

The final term is trivially positive. Note that the last step follows from the fact that $H_{1,2} = H_{2,1}$. $\square$

References

[1] Meinolf Blawat, Klaus Gaedke, Ingo Huetter, Xiao-Ming Chen, Brian Turczyk, Samuel Inverso, Benjamin W Pruitt, and George M Church. Forward error correction for DNA data storage. *Procedia Computer Science*, 80:1011–1022, 2016.

[2] James Bornholt, Randolph Lopez, Douglas M Carmean, Luis Ceze, Georg Seelig, and Karin Strauss. A DNA-based archival storage system. *ACM SIGARCH Computer Architecture News*, 44(2):637–649, 2016.

[3] Joshua Brakensiek, Venkatesan Guruswami, and Samuel Zbarsky. Efficient low-redundancy codes for correcting multiple deletions. *IEEE Transactions on Information Theory*, 64(5):3403–3410, 2018.

[4] Boris Bukh, Venkatesan Guruswami, and Johan Håstad. An improved bound on the fraction of correctable deletions. *IEEE Transactions on Information Theory*, 63(1):93–103, 2017.

[5] Kuan Cheng, Venkatesan Guruswami, Bernhard Haeupler, and Xin Li. Efficient linear and affine codes for correcting insertions/deletions. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1–20, 2021.

[6] Kuan Cheng, Bernhard Haeupler, Xin Li, Amirbehshad Shahrasbi, and Ke Wu. Synchronization strings: highly efficient deterministic constructions over small alphabets. In *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2185–2204, 2019.

[7] Kuan Cheng, Zhengzhong Jin, Xin Li, and Ke Wu. Deterministic document exchange protocols, and almost optimal binary codes for edit errors. In *Proceedings of the 59th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 200–211, 2018.

[8] Kuan Cheng, Zhengzhong Jin, Xin Li, and Ke Wu. Block edit errors with transpositions: Deterministic document exchange protocols and almost optimal binary codes. In *Proceedings of the 46th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 132 of *LIPIcs*, pages 37:1–37:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[9] Mahdi Cheraghchi and João Ribeiro. An overview of capacity results for synchronization channels. *IEEE Transactions on Information Theory*, 2020.

[10] George M Church, Yuan Gao, and Sriram Kosuri. Next-generation digital information storage in DNA. *Science*, 337(6102):1628–1628, 2012.

[11] Tai Do Duc, Shu Liu, Ivan Tjuawinata, and Chaoping Xing. Explicit constructions of two-dimensional reed-solomon codes in high insertion and deletion noise regime. *IEEE Transactions on Information Theory*, 67(5):2808–2820, 2021.

[12] Nick Goldman, Paul Bertone, Siyuan Chen, Christophe Dessimoz, Emily M LeProust, Botond Sipos, and Ewan Birney. Towards practical, high-capacity, low-maintenance information storage in synthesized DNA. *Nature*, 494(7435):77, 2013.

[13] Venkatesan Guruswami. *List decoding of error-correcting codes: winning thesis of the 2002 ACM doctoral dissertation competition*, volume 3282. Springer Science & Business Media, 2004.

[14] Venkatesan Guruswami, Bernhard Haeupler, and Amirbehshad Shahrasbi. Optimally resilient codes for list-decoding from insertions and deletions. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 524–537, 2020.

[15] Venkatesan Guruswami, Xiaoyu He, and Ray Li. The zero-rate threshold for adversarial bit-deletions is less than 1/2. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 727–738, 2022.

[16] Venkatesan Guruswami and Ray Li. Efficiently decodable insertion/deletion codes for high-noise and high-rate regimes. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 620–624, 2016.

[17] Venkatesan Guruswami and Carol Wang. Deletion codes in the high-noise and high-rate regimes. *IEEE Transactions on Information Theory*, 63(4):1961–1970, 2017.

[18] Bernhard Haeupler. Optimal document exchange and new codes for insertions and deletions. In *Proceedings of the 60th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 334–347, 2019.

[19] Bernhard Haeupler, Aviad Rubinstein, and Amirbehshad Shahrasbi. Near-linear time insertion-deletion codes and $(1+\varepsilon)$-approximating edit distance via indexing. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 697–708, 2019.

[20] Bernhard Haeupler and Amirbehshad Shahrasbi. Synchronization strings: Explicit constructions, local decoding, and applications. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 841–854, 2018.

[21] Bernhard Haeupler and Amirbehshad Shahrasbi. Synchronization strings and codes for insertions and deletions–a survey. *IEEE Transactions on Information Theory*, 2021.

[22] Bernhard Haeupler and Amirbehshad Shahrasbi. Synchronization strings: Codes for insertions and deletions approaching the Singleton bound. *Journal of the ACM (JACM)*, 68(5):1–39, 2021.

[23] Bernhard Haeupler, Amirbehshad Shahrasbi, and Madhu Sudan. Synchronization strings: List decoding for insertions and deletions. In *Proceedings of the 45th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 107 of *LIPIcs*, pages 76:1–76:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

[24] Bernhard Haeupler, Amirbehshad Shahrasbi, and Ellen Vitercik. Synchronization strings: Channel simulations and interactive coding for insertions and deletions. In *Proceedings of the 45th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 107 of *LIPIcs*, pages 75:1–75:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

[25] Tomohiro Hayashi and Kenji Yasunaga. On the list decodability of insertions and deletions. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 86–90, 2018.

[26] Vladimir I Levenshtein. Elements of coding theory. *Diskretnaya matematika i matematicheskie voprosy kibernetiki*, pages 207–305, 1974.

[27] Shu Liu, Ivan Tjuawinata, and Chaoping Xing. On list decoding of insertion and deletion errors. *CoRR*, abs/1906.09705, 2019.

[28] Shu Liu, Ivan Tjuawinata, and Chaoping Xing. Efficiently list-decodable insertion and deletion codes via concatenation. *IEEE Transactions on Information Theory*, 67(9):5778–5790, 2021.

[29] Hugues Mercier, Vijay K Bhargava, and Vahid Tarokh. A survey of error-correcting codes for channels with symbol synchronization errors. *IEEE Communications Surveys & Tutorials*, 12(1):87–96, 2010.

[30] Michael Mitzenmacher. A survey of results for deletion channels and related synchronization channels. *Probability Surveys*, 6:1–33, 2009.

[31] Lee Organick, Siena Dumas Ang, Yuan-Jyue Chen, Randolph Lopez, Sergey Yekhanin, Konstantin Makarychev, Miklos Z Racz, Govinda Kamath, Parikshit Gopalan, Bichlien Nguyen, et al. Scaling up DNA data storage and random access retrieval. *BioRxiv*, page 114553, 2017.

[32] Antonia Wachter-Zeh. List decoding of insertions and deletions. *IEEE Transactions on Information Theory*, 64(9):6297–6304, 2018.

[33] SM Hossein Tabatabaei Yazdi, Han Mao Kiah, Eva Garcia-Ruiz, Jian Ma, Huimin Zhao, and Olgica Milenkovic. DNA-based storage: Trends and methods. *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, 1(3):230–248, 2015.