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

Bernhard Haeupler
Carnegie Mellon University and *ETH Zurich*
Zurich, Switzerland
bernhard.haeupler@inf.ethz.ch

Amirbehshad Shahrasbi
Microsoft
Redmond, WA, USA
ashahrasbi@microsoft.com

2022 IEEE Information Theory Workshop (ITW) | 978-1-6654-8341-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/ITW54588.2022.9965935

***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.**

***Index Terms*—Coding for Insertions and Deletions, List Decoding, Synchronization, Error-Resilience, List-Decoding Capacity**

I. 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], [23], [25], [29] and practical perspectives [1], [2], [10], [12], [32], [34]. 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 $R = \lim_{n\to\infty} \frac{\log_q |C_n|}{n}$.

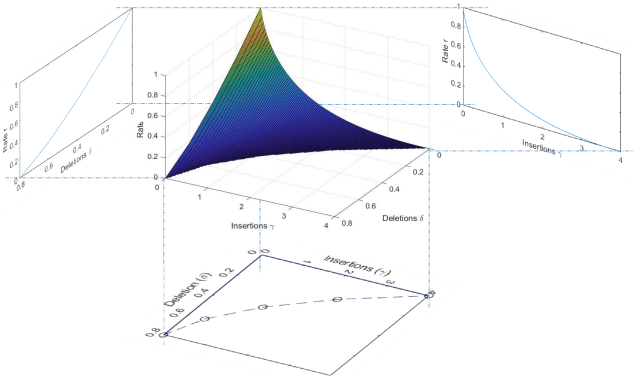

Fig. 1: Depiction of our outer bound for $q = 5$ and its three projections for the insertion-only case (on the right), the deletion-only case (on the left), and the zero-rate case (on the bottom). The projection graphs are exactly matching the state-of-the-art results of [14], [24].

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

Bernhard Haeupler's work is supported in part by NSF grants CCF-1527110, CCF-1618280, CCF-1814603, CCF-1910588, NSF CAREER award CCF-1750808, a Sloan Research Fellowship, and funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (ERC grant agreement 949272). Amirbehshad Shahrasbi's work is supported in part by CRA Computing Innovation Postdoctoral Fellowship. A full version of this paper is available at [21].

Section I-B) 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 1 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 1 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.

A. 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 any given γ, δ, and q). The novel outer bounds we prove are given in the 3D-plot of Figure 1 for an alphabet size of $q = 5$ (similar plots for any given q apply). We develop these outer bounds in Section II. (See Theorem II.4)

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 [24]. Similarly, we match the best insertion-only bounds known (also from [24]) 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 1 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.

Lastly, for the sake of completeness and as a comparison point, we also provide a general inner bound in Section III. 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. 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 [24] 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.

B. 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], [24], [26], [29], [33]. 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], [22], [30], [31], which give detailed accounts of such works.

As noted above, with the exception of [29], 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) List-Decodable Insdel Codes Over Large Constant-Size Alphabets: The rate-distance tradeoff for list-decodable error correcting codes has been studied in [24] 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 [23], [24] 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 [24] 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 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.

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"*. 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], [26] 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 [33] gave Johnson-type bounds on list-decodability and list-sizes of codes given their minimum edit-distance. In 2018, Hayashi and Yasunaga [26] made corrections to the results presented in [33], 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 1).

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. [24]. These projections are shown in Figure 1 to the right and left respectively. The inner bounds presented in [24] are derived by analyzing list-decoding properties of random codes. Here, we briefly review (the ideas of) the outer bounds from [24] as these will be helpful for the remainder of this paper.

a) 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 $a \in [q]$. [24] 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, [24] provides a piece-wise linear outer bound for all values of $0 < \delta < 1 - \frac{1}{q}$.

b) 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, [24] used the probabilistic method: For a given codeword $x \in [q]^n$, [24] 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.

4) General Case: Liu et al. [28] 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 [28] were discovered, several correctness issues with key approaches of [28] for outer bounds were identified. These results have been removed in [29], the final version of [28]. The underlying issues seem hard to fix without substantially new ideas, as also reported in the acknowledgements of [29]. As a result, [29] 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 $q \geq 3$. (See the full version for a proof.)

II. Outer Bound

We start by reminding the following outer bound for the insertion-only case from [24].

Theorem II.1 (From [24])**.** *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 II.1 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. (a multiple of $\frac{1}{q}$) This outer bound forms the backbone of our final result.

Theorem II.2. *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 II.1 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 $\binom{q}{q-d} q^{n(1-\delta)\left[\log_q q' - \log_q(\gamma'+1)-\gamma'\left(\log_q\frac{\gamma'+1}{\gamma'}-\log_q\frac{q'}{q'-1}\right)\right]}$ which implies the theorem statement. □

Given the nice and explicit form of Theorem II.2 for any q and γ with multiple specific values of δ, it seems tempting to conjecture that the restriction of δ is unnecessary making $(1-\delta)\left[\left(1+\frac{\gamma}{1-\delta}\right)\log_q\frac{q-d}{\frac{\gamma}{1-\delta}+1} - \frac{\gamma}{1-\delta}\cdot\left(\log_q\frac{q-d-1}{\frac{\gamma}{1-\delta}}\right)\right]$ 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 II.2, 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 II.2 hits zero at exactly the corner points of the piece-wise linear resilience region F_q characterized by [14]. Taking this as an inspiration, one could try to extend the bound of Theorem II.2 to all values of δ by considering for each q and each rate r the roughly $\frac{q}{r}$ points where Theorem II.2 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 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 II.2 and in particular combines points with different rates, also holds and is needed to give our final outer bound.

Theorem II.3. *For a fixed q, suppose that $(\gamma_0, \delta_0 = \frac{d_0}{q})$ and $(\gamma_1, \delta_1 = \frac{d_1}{q})$ are two error rate combinations for which Theorem II.2 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 has a rate of at most r.*

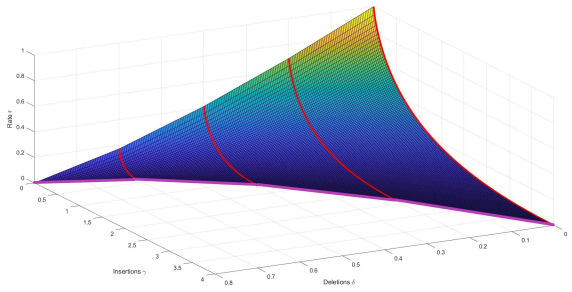

Fig. 2: Outer bound for rate for $q = 5$. The special case where $\delta = \frac{d}{q}$ for some integer d is indicated with red lines.

See Figure 2 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 II.2.

Theorem II.3 together with Theorem II.2 gives a conceptually very clean description of our outer bound. However, an (exact) evaluation of the outer bound as given by Theorem II.3 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 II.3, 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 II.5 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 II.4. *Let C be a q-ary insertion-deletion code that is list-decodable from $\gamma \in [0, q-1]$ fraction of insertions and $\delta \in [0, 1-\frac{1}{q}]$ fraction of deletions. Then, the rate of C is no larger than* $\alpha\left(1-\frac{d}{q}\right)\left((1+\gamma_0)\log_q\frac{q-d}{1+\gamma_0} - \gamma_0\log_q\frac{q-d-1}{\gamma_0}\right) + (1-\alpha)\left(1-\frac{d+1}{q}\right)\left((1+\gamma_1)\log_q\frac{q-d-1}{1+\gamma_1} - \gamma_1\log_q\frac{q-d-2}{\gamma_1}\right)$ *for $d = \lfloor\delta q\rfloor$, $\alpha = 1-\delta q+d$, and all $\gamma_0,\gamma_1 \ge 0$ 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 II.5.*

Proof of Theorems II.3 and II.4. We first note that the statements of Theorem II.4 and Theorem II.3 are merely a rephrasing of each other with the exception that Theorem II.4 only allows and optimizes over convex combinations of neighboring spokes of Theorem II.2, 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 $\left\{(\gamma,\delta) \mid \delta = \frac{d}{q}, 0 \le d \le q-1, d \in \mathbb{Z}\right\}$, Theorem II.2 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. (proof in [21].) Any value given as a convex combination between two non-neighboring spokes can therefore be at least matched (and actually improved due to

strict convexity) by choosing a different convex combination between neighboring spokes. This justifies the "restricted" formulation of Theorem II.4, which helps in reducing the number of parameters and simplifies calculations.

In order to prove Theorem II.4 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 $\Sigma_0, \Sigma_1 \subseteq [q]$, $q_0 = |\Sigma_0| = q-d$, $q_1 = |\Sigma_1| = q-d-1$.

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 thus $|D_0| \geq \frac{|C_0|}{\mathsf{poly}(n)}$.

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 [24], 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 describe members of D_0 like y as the concatenation (y_0, y_1) where $|y_0| = n_0 = n\alpha(1-\frac{d}{q})$ and $|y_1| = n_1 = n(1-\alpha)(1-\frac{d+1}{q})$.

We have $\mathbb{E}[X] = \sum_{y=(y_0,y_1)\in D_0} \Pr\{y \text{ is a subseq. of } Z\}$. Note that $\Pr\{y \text{ is a subseq. of } Z\}$ is not smaller than $\prod_{j=0,1} \Pr\{y_j \text{ is a subseq. of } Z_j\}$. Also, conditioning on the leftmost occurrence of Z_j in y_j, we can write down the $\Pr\{y_j \text{ is a subseq. of } Z_j\}$ as $\sum_{1\leq a_1<\cdots<a_{n_j}\leq n_j(1+\gamma_j)} \frac{1}{|\Sigma_j|^{n_j}} \left(1-\frac{1}{|\Sigma_j|}\right)^{a_{n_j}-n_j}$. We use this expression in [21] to bound below $\mathbb{E}[X]$ by $L = |D_0| q^{\sum_{j=0,1} n_j \left(\gamma_j \log_q \frac{q_j-1}{\gamma_j} - (1+\gamma_j)\log_q \frac{q_j}{1+\gamma_j} + o(1)\right)}$. This means that there exists some realization of Z to which at least L codewords of $\mathcal{C}$ are subsequences. In order for C to be list-decodable, this quantity needs to be sub-exponential. Therefore, $r_C = \frac{\log_q |D_0| + O(1)}{n} \leq \sum_{j=0,1} \frac{n_j}{n} \left((1+\gamma_j)\log_q \frac{q_j}{1+\gamma_j} - \gamma_j \log_q \frac{q_j-1}{\gamma_j}\right)$. This leads to the upper bound stated in Theorem II.4 for r_C. (see the full version [21] for a complete calculation and proof.) □

Theorem II.5. *The optimal choice for γ_0 in Theorem II.4 satisfies $(1+1/\gamma_1)(1-1/(q-d-1)) = (1+1/\gamma_0)(1-1/(q-d))$. Together with the equation $\alpha(1-\frac{d}{q})\gamma_0 + (1-\alpha)(1-\frac{d+1}{q})\gamma_1 = \gamma$, this gives an explicit expression for γ_0 in terms of q,γ, $d = \lfloor \delta q \rfloor$, and $\alpha = 1-\delta q+d$ which can be found in the full version [21].*

III. 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. 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 for deletions and combination of insertions and deletions.

Lemma III.1 (From [27])**.** *Let n, n_i, and q be positive integers and $S \in [q]^n$. Then, $|\mathcal{B}_i(S, n_i)| = \sum_{i=0}^{n_i} \binom{n+n_i}{i}(q-1)^i$.*

Lemma III.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 $q^{n\left(H_q(\delta)+(1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right)-\delta\log_q(q-1)\right)+o(n)}$ where $H_q(\cdot)$ denotes the q-ary entropy function defined as $H_q(x) = x\log_q(q-1) - x\log_q x - (1-x)\log_q(1-x)$.*

Proof Sketch (Full proof available in [21]).

$$
\begin{aligned}
|\mathcal{B}(x, \gamma n, \delta n)| &\leq \sum_{x_0 \in \mathcal{B}_d(x, \delta n)} |\mathcal{B}_i(x_0, \gamma n)| \\
&\leq \binom{n}{\delta n} \sum_{i=0}^{\gamma n} \binom{n(1-\delta+\gamma)}{i} (q-1)^i \qquad (1) \\
&\leq n\gamma \binom{n}{n\delta} \binom{n(1-\delta+\gamma)}{n\gamma} (q-1)^{\gamma n} \qquad (2) \\
&= q^{n\left(H_q(\delta)+(1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right)-\delta\log_q(q-1)\right)+o(n)}
\end{aligned}
$$

Note that (1) follows from Lemma III.1 and (2) is true because the term in summation reaches its maximum when $i = n\gamma$. □

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 III.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 $R < 1-(1-\delta+\gamma)H_q\left(\frac{\gamma}{1-\delta+\gamma}\right) - H_q(\delta) + \gamma\log_q(q-1)$ is list-decodable from γn insertions and δn deletions WHP.*

Proof Sketch (Full proof in [21]). We use Lemma III.2 to bound the probability of a fixed string falling within a certain ball around a codeword of a random code. We then bound above the probability of such string being close to $l+1$ codewords, i.e., violating the l-list-decodability condition. Taking an upper bound over all such center strings, we bound above the probability of a random code not being list-decodable and find a range for R where such probability is negligible. □

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. Rate-distance trade-offs for list-decodable insertion-deletion codes. *arXiv preprint arXiv:2009.13307*, 2020.

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

[23] 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.

[24] 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.

[25] 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.

[26] 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.

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

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

[29] 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.

[30] 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.

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

[32] 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.

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

[34] 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.