On Transmission Over Deletion Channels

Suhas N. Diggavi Matthias Grossglauser
AT&T Shannon Laboratory
180 Park Avenue, Bldg 103
Florham Park NJ 07932
{suhas,mgross}@research.att.com

Abstract

In this paper, we develop lower bounds on the achievable rate for deletion channels. Deletion channels occur when symbols are randomly dropped, and a subsequence of the transmitted symbols is received. In deletion channels, unlike erasure channels, there is no side-information about which subsequence is received. We show that the achievable rate in deletion channels differs from that of erasure channels by at most $H_0(p_d) - p_d \log \frac{K}{K-1}$ bits, where p_d is the deletion probability, K is the alphabet size and $H_0(\cdot)$ is the binary entropy function. We also develop lower bounds for the binary deletion channel that improve the bounds reported in the literature.

1 Introduction

In a packet-switched communication network, such as the Internet, the source of a session encodes information in a set of packets, which are transported as independent units through a set of links to reach their destination. A packet reaches its destination if there exists a route to the destination, and if there is buffer space available at every node along the path followed by this packet. In this paper, we analyze the capacity of a channel that models the basic ingredients of packet-switched networks in "normal" operation: (a) a packet either reaches its destination or is lost completely, and (b) the original order of packets is conserved.

In previous work [4], we had modeled this channel as an erasure channel, where the destination receives either the exact original packet, or an erasure symbol E. The focus in the erasure model was on the memory of the channel. As the erasure process models packet loss, which is due to temporary congestion at a finite-buffer queue in the network, the erasure process can potentially possess a complex memory structure. We proved that for a very large class of erasure processes, the channel capacity of the erasure channel is equal to $\log(K)(1 - p_E)$, where $\log(K)$ bits is the packet size (hence the alphabet size is K) and p_E is the long-term average of packets lost.

The main limitation of the erasure channel as a model for a finite-buffer queue (or a sequence thereof) is that there is no mechanism in the network to explicitly signal a dropped packet to the destination. Rather, transport protocols such as TCP use *sequence numbers* in the packet header[1] to detect lost packets. Packets sent by the source are numbered sequentially. The destination can infer the loss of one or several packets from the sequence number of the packet following the loss. The sequence number uses up a certain number of bits (32 bits in the case of TCP/IP) to detect lost packets and to request retransmission of those packets. We can view it as a code that converts the deletion channel into an erasure channel. A fundamental question therefore arises: if we do not assume *a-priori* the existence of sequence numbers, what is the capacity of the resulting channel? This question naturally leads to the *deletion channel*, which essentially differs from the erasure channel in that the destination receives no

[1] strictly speaking, in the TCP header.

explicit symbol indicating loss of a packet. Instead, the received sequence of symbols is shorter than the original sequence, with deleted symbols simply removed.

The deletion channel is a special case of insertion/deletion/substitution channels[2] which model the effect of synchronization errors and have a long history [8, 7, 10, 5, 11, 9]. A coding theorem for such channels in terms of maximizing mutual information over input distributions was proved in [5]. However, even in the presence of memoryless deletions, this does not lead to a single-letter characterization for achievable rates. Gallager, in an unpublished report [7], analyzes the performance of convolutional codes over insertion/deletion/substitution channels. He proposed adding a pseudo-random sequence to convolutional codes to correct such synchronization errors and derived lower bounds for achievable rates using sequential decoding for these codes. A similar idea was studied in "watermarking" codes proposed in [3], where LDPC codes and iterative decoding were used. Zigangirov [11] studied a more general insertion/deletion channel and derived improved lower bounds (compared to [7]) to the performance of convolutional codes with sequential decoding. The bounds of [7, 11] coincide for memoryless deletion channels and are given by

$$C_{del} \geq 1 - H_0(p_d), \quad p_d \leq 0.5 \tag{1}$$

where $H_0(p_d) = -(1-p_d)\log(1-p_d) - p_d \log(p_d)$ is the binary entropy function. In Section 3.1, we provide an alternative proof for this result. Our proof also yields lower bounds for larger (non-binary) alphabet sizes and when the deletion process is stationary and ergodic. In Section 3.2 we derive bounds that improve (1) by using codebooks with memory. Ullmann [10] pursues a combinatorial approach to derive upper and lower bounds for the binary insertion/deletion channel. His bounds are (see (33) and (44) in [10])

$$1 - p\log_2 e^2 \left(\frac{3}{2p} + \frac{15}{16}\right)\left(\frac{3}{2p} + \frac{47}{16}\right) \leq C \leq 1 - (1+p)\log_2(1+p) + p\log_2(2p) \tag{2}$$

where p in his notation is the fraction of total insertion/deletion errors asymptotically in the codeword block size. The lower bound (1) in [7, 11] are sharper than (2). However, (2) apparently provides the only known upper bound to insertion/deletion channels when asymptotically (in the codeword block size) the number of synchronization errors is a non-zero fraction of the codeword block size.

The insertion/deletion/substitution channel was also pioneered by Levenshtein [8], where asymptotic bounds in the number of codewords capable of correcting up to a *finite* number of synchronization errors was studied. He also provided number-theoretic constructions for such codes and motivated a large body of literature on this topic (see [9] for a recent survey of such code constructions). Our focus in this paper is asymptotic information-theoretic bounds when the number of deletions is a non-zero fraction of the codeword block size and not on code constructions.

The remainder of this paper is structured as follows. Section 2 formally states the problem. In Section 3, we present the main results of the paper, which are lower bounds on the transmission rates over deletion channels. Section 4 concludes the paper with numerical results and a discussion of several open issues.

2 Problem Statement

In this paper, we consider the capacity of the K-ary deletion channel, which is defined as follows. Let $x = (x_1, \ldots, x_n)$ be a codeword, where $x_i \in \{1, \ldots, K\}$. A *deletion pattern* d is a binary vector $(d_1, \ldots, d_n)$, where $d_i = 1$ indicates that the i-th symbol of x is deleted, and $d_i = 0$ indicates that the i-th symbol is received at the output. We are mainly interested in i.i.d. distributions (with $\mathbb{P}\{D_i = 1\} = p_d$) for the binary sequence D_i, but results in Section 3.1 also apply when D is stationary and ergodic.

[2] In an insertion channel, additional symbols can randomly be inserted into the codeword. Substitutions are the familiar symbol errors for noisy channels.

Given an input sequence $x = (x_1, \ldots, x_n)$ and a deletion sequence $d = (d_1, \ldots, d_n)$, the output sequence $y = d \circ x$ is formally defined as follows. Let e be the total number of 1's in d, *i.e.*, the number of deletions. We define $i(k)$ as the position of the k-th 0 in the sequence d; clearly $0 \leq k \leq n - e$. Then the received sequence is $y = (y_1, \ldots, y_{n-e})$, with $y_k = x_{i(k)}$, $1 \leq k \leq n - e$. In other words, y is a sequence of length $n - e$ containing the non-deleted symbols in x. Note that the deletion channel has memory in that $p(y|x)$ does not become a product distribution even for a i.i.d. deletion process. Intuitively, this is because the k-th output symbol y_k depends on the entire history of the deletion process up to $i(k)$.

We define a $(|\mathcal{C}|, n)$ code as a set of $|\mathcal{C}|$ codewords $\mathcal{C} = \{x(1), \ldots, x(|\mathcal{C}|)\}$ of length n. The encoding function results in a codeword $x(i)$ to be sent when a message $i \in \{1, \ldots, |\mathcal{C}|\}$ is drawn. The deletion process D causes the random sequence $Y = y$ of length at most n to be received. We also define a decoding function $\hat{W} : \mathcal{S} \to \{1, \ldots, |\mathcal{C}|\}$, where $\mathcal{S}$ is the set of all K-ary sequences of length of at most n. The average probability of error for a given codebook $\mathcal{C}$ and decoding function $\hat{W}$ is defined as

$$P_e(\mathcal{C}, \hat{W}) = \frac{1}{|\mathcal{C}|} \sum_{i=1}^{|\mathcal{C}|} \mathbb{P}\left\{\hat{W}(Y) \neq i | X = x(i)\right\}. \tag{3}$$

We define a rate R to be achievable if there exists a sequence of $(\lceil 2^{nR} \rceil, n)$ codes and a decoding rule $\hat{W}$ such that the average probability of error $P_e(\mathcal{C}, \hat{W})$ tends to 0 as $n \to \infty$.

We introduce the following notation. We denote by $|x|$ the length of a sequence x. We call the *weight* of a binary sequence the number of 1's in that sequence. For a codeword x and a deletion pattern d, we denote the output symbol y as $y = d \circ x$. We define the derivative Δx of a sequence x as a sequence of length $|x| - 1$, whose l^{th} component is 1 if $x_{l+1} \neq x_l$, and 0 otherwise.

3 Bounds on the deletion channel

In this section, we develop lower bounds for the deletion channel formulated in Section 2. In Section 3.1, we develop a simple lower bound based on i.i.d. codebooks. This bound is valid even when there is memory in the deletion process, *i.e.*, when the deletion process is just assumed to be stationary and ergodic. We refine the lower bound in Section 3.2 using input codebooks with memory. This improved bound is valid only for i.i.d. deletion patterns. However, a bound using this approach can also be worked out when the deletion patterns are Markovian.

3.1 First lower bound

We assume a stationary and ergodic model for the deletion process D. Therefore, for large n, the fraction of deleted packets is close to $(1 - \theta) \stackrel{def}{=} p_d$ with high probability. Note that, as mentioned in Section 1, if the deletion patterns were known (through sequence numbers for example) then the channel would be equivalent to an erasure channel, whose capacity is $\theta \log(K)$. Clearly, conveying the deletion pattern to the receiver constitutes side-information and therefore this rate is an upper bound to the deletion channel capacity.

To study this problem we use the following lemma proved in [1] for longest common subsequences of random sequences.

Lemma 3.1 *[1] For a given K-ary sequence s of length $|s|$, the number $F(n, s, K)$ of K-ary sequences of length n which contain sequence s as a subsequence is given by:*

$$F(n, |s|, K) = \sum_{j=|s|}^{n} \left(\begin{array}{c} n \\ j \end{array} \right) (K-1)^{n-j} \tag{4}$$

Note that the function $F(\cdot)$ depends on s *only* through its length $|s|$. Note that Lemma 3.1 implies that if X is an i.i.d. sequence with uniform distribution over its K-ary alphabet then,

$$\mathbb{P}\{s \text{ subsequence of } X\} = \frac{F(n, |s|, K)}{K^n} \tag{5}$$

as all sequences are equally likely. Given this result, we prove the following lower bound on the capacity of the deletion channel.

Theorem 3.2 *Given a stationary and ergodic deletion channel with long-term deletion probability given by $p_d = 1 - \theta$ (with $p_d < 1 - 1/K$), and an input alphabet size K, the capacity of this channel is lower bounded as*

$$C_{del} \geq \log\left(\frac{K}{K-1}\right) + \theta \log(K-1) - H_0(\theta), \tag{6}$$

where $H_0(\theta) = -(1-\theta)\log(1-\theta) - \theta\log(\theta)$ is the binary entropy function.

Proof: Let us generate a random codebook of 2^{nR} i.i.d. codewords chosen uniformly from a K-ary alphabet. As the channel randomly deletes symbols from a codeword, the length of the received sequence $M = |Y|$ is a random variable. Assuming that the deletion process is stationary and ergodic, it follows that

$$\lim_{n\to\infty} \mathbb{P}\left\{\left|\frac{M}{n} - \theta\right| > \epsilon\right\} \to 0. \tag{7}$$

We use the following decoding rule. If the received sequence Y has fewer than $m = (\theta-\epsilon)n$ symbols, we declare an error. Because of (7), the probability of this error goes to zero. If the received sequence Y has at least m symbols, we check the number of codewords in the codebook that could have produced Y under any deletion pattern, i.e., the codewords that contain Y as a subsequence. If there is more than one possible codeword, then we declare a *collision error*. If there is no collision, we are certain that the unique candidate codeword is the correct one (as the channel is not noisy), and the transmission is successful.

We now compute the asymptotic probability of collision error. We first consider the *pairwise* probability of collision error between two codewords x_1 and x_2 from the codebook, averaged over all the codebooks. A collision error occurs between two codewords x_1 and x_2 if the received sequence $Y = D_1 \circ x_1$ generated by a random deletion pattern D_1 is a subsequence of x_2 (because this implies that there exists a deletion pattern d_2 such that $d_2 \circ x_2 = Y$.)

Consider the error probability conditional on the number of received symbols M. This probability obviously decreases with M, because the probability of a common subsequence decreases with its length. Therefore, an upper bound for the collision probability can be obtained by setting $m = \lfloor(\theta-\epsilon)n\rfloor$ and assuming $M = m$. For computational reasons, we set $m = (\theta-\epsilon)n - 1$, which is conservative. Therefore, we can write the pairwise error probability that x_2 collides with the transmitted codeword x_1 averaged over random codebooks as,

$$\begin{aligned}
\bar{P}_2 &= \mathbb{E}_{\mathcal{C}}[\mathbb{P}\{\text{error}|x_1, m\}] = \sum_{y,|y|=m} \mathbb{P}\{y \text{ is a subsequence of } X_2|X_1\}\,\mathbb{P}\{y = D \circ X_1\} \qquad (8)\\
&\overset{(a)}{=} \frac{F(n,m,K)}{K^n} \sum_{y,|y|=m} \mathbb{P}\{y = D \circ x_1\} \overset{(b)}{=} \frac{F(n,m,K)}{K^n}\\
&\overset{(c)}{\leq} \frac{1}{K^n} n \begin{pmatrix} n \\ m \end{pmatrix} (K-1)^{n-m} \overset{(d)}{\leq} \frac{1}{K^n} n 2^{nH_0(\frac{m}{n})} (K-1)^{n-m}
\end{aligned}$$

where (a) follows by using (5) and the fact that X_1 and X_2 are independent of each other; (b) follows because $\sum_{y,|y|=m} \mathbb{P}\{y = D \circ x_1\} = 1$, as the probability summed over all deletion

patterns (conditioned on the weight of the deletion pattern) is unity; (c) follows from the inequality (which can be easily verified [1]) that $F(n,m,K) \leq n \left(\begin{array}{c} n \\ m \end{array} \right) (K-1)^{n-m}$; and (d) follows from the inequality $\left(\begin{array}{c} n \\ m \end{array} \right) \leq 2^{nH_0(\frac{m}{n})}$ (see [2], Chapter 12, pp 284).

Now, we use a union bound over all codewords x_2 to bound the error probability $\bar{P}_e$ (averaged over codebooks) as,

$$\bar{P}_e \leq 2^{nR}\mathbb{E}_{\mathcal{C}}[\mathbb{P}\{\text{error}|x_1, M > m\}] + \mathbb{P}\{M \leq m\} \leq n \left[\frac{(K-1)2^R 2^{H_0(\frac{m}{n})}}{K(K-1)^{\frac{m}{n}}} \right]^n + \delta_n. \quad (9)$$

Therefore, if the first term decreases exponentially with n, the probability of error goes to zero asymptotically in n as $\delta_n \to 0$ from (7). This happens when

$$R < \log\left(\frac{K}{K-1}\right) + \theta \log(K-1) - H_0(\theta) \quad (10)$$

Therefore, by using the well-known random coding argument [2], there exists a deterministic codebook which has an achievable rate given by R. Note that this is in the regime where θ is such that $\log(\frac{K}{K-1}) + \theta\log(K-1) - H_0(\theta) > 0$, which occurs for $p_d < 1 - 1/K$. ■

Therefore, as the capacity of the deletion channel is upper bounded by that of the erasure channel, we obtain the following double inequality,

$$\log\left(\frac{K}{K-1}\right) + \theta\log(K-1) - H_0(\theta) \leq C_{del} \leq \theta\log(K) \quad (11)$$

For the binary case ($K = 2$), Theorem 3.2 coincides with [7, 11].

Corollary 3.3 *Given a stationary and ergodic deletion channel with long term deletion probability given by $1-\theta$ (with $\theta > 1/2$), and a binary input alphabet, the capacity of this channel is lower bounded as*

$$C_{del} \geq 1 - H_0(\theta), \quad (12)$$

where $H_0(.)$ is the binary entropy function.

3.2 Markov lower bound

In Section 3.1, the codewords were i.i.d. However, we believe that the optimal codebook construction has memory, because the deletion channel has memory. In this section, we sharpen the result in Section 3.1 by using input codebooks which are generated from a Markov process. For simplicity, we consider only first-order Markov chains.

In analogy with the analysis for i.i.d. codewords in the previous section, we need to find the probability that two independent Markov chains of length n have a common subsequence of length greater than m. In order to simplify the analysis, we assume that the deletion sequences are i.i.d. processes[3]. Moreover, in this section, we focus on the binary alphabet case; extending the technique to larger alphabets is straightforward.

Theorem 3.4 *Given an i.i.d. deletion channel with deletion probability given by $p_d = 1-\theta$, and a binary input alphabet, the capacity of this channel is lower bounded as*

$$C_{del} \geq \sup_{\substack{\gamma>0 \\ 0<p<1}} \left[-\theta\log\{(1-q)A + qB\} - \gamma\right] \text{ nats} \quad (13)$$

where $A = A(p,\gamma) = \frac{(1-p)e^{-\gamma}}{(1-pe^{-\gamma})}$, $B = B(p,\gamma) = \frac{(1-p)^2e^{-2\gamma}}{(1-pe^{-\gamma})} + pe^{-\gamma}$ and $q = q(p) = 1 - \frac{1-p}{1+(1-\theta)(1-2p)}$.

[3]The technique developed here can be easily extended to the case where the deletion sequences are finite state Markov processes as well.

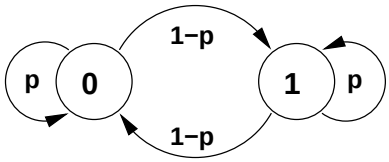

Figure 1: Markov chain generating codebook.

Proof: We generate a random codebook $\mathcal{C}$ of e^{nR} codewords, with each codeword an independent sequence generated using the first-order Markov chain illustrated in Figure 1. We use the same decoding rule as the one used in the proof of Theorem 3.2, *i.e.*, we declare an error either when the received sequence Y has fewer than $m = (\theta - \epsilon)n$ symbols, or when there is more than one codeword in $\mathcal{C}$ that contains Y as a subsequence. In order to compute the asymptotic collision error probability, we again consider the *pairwise probability of error* $\bar{P}_2$ between two codewords x_1 and x_2 averaged over random codebooks. As in (8) this can be written as,

$$\bar{P}_2 = \mathbb{E}_{\mathcal{C}}[\mathbb{P}\{\text{error}|X_1, m\}] = \sum_{y,|y|=m} \mathbb{P}\{y \text{ is a subsequence of } X_2|x_1\}\,\mathbb{P}\{y = D \circ X_1\} \quad (14)$$

The central difference between the calculation done in (8) and here is that when the input codebook has memory, the term $\mathbb{P}\{y \text{ is a subsequence of } X_2|x_1\}$ depends explicitly on the subsequence y not just through its length m as was the case in (8). Moreover, this implies that we need to also explicitly calculate the term $\mathbb{P}\{y = D \circ x_1\}$ when the deletion process is i.i.d. and x_1 is a Markov process. For both these calculations, the weight of the derivative Δy of y becomes important. Note that the number N_i of binary sequences s of length $|s| = m$ with Δs having weight i is given by

$$N_i = 2\begin{pmatrix} m-1 \\ i \end{pmatrix} \quad (15)$$

Therefore, to calculate the pairwise error probability expression in (14) we prove the following results:

Result 1: The probability of a given subsequence y occurring through an i.i.d. deletion process D in X is given by

$$\mathbb{P}\{y = D \circ X\} = \frac{1}{2}(1-q)^i q^{m-1-i}, \quad (16)$$

where i is the weight of Δy, and $q = 1 - \frac{1-p}{1+(1-\theta)(1-2p)}$.

Result 2: Probability of a given subsequence y occurring in a Markov sequence X_2 generated by the transition probability illustrated in Figure 1 is bounded by,

$$\mathbb{P}\{y \text{ is a subsequence of } X_2\} \leq \inf_{\gamma>0} F e^{\gamma n} A^i B^{m-1-i}, \quad (17)$$

where $A = \frac{(1-p)e^{-\gamma}}{(1-pe^{-\gamma})}$, $B = \frac{(1-p)^2 e^{-2\gamma}}{(1-pe^{-\gamma})} + pe^{-\gamma}$, and $F = \frac{1}{2}e^{-\gamma}\left[1 + \frac{(1-p)e^{-\gamma}}{(1-pe^{-\gamma})}\right]$. Again, in (17) i denotes the weight of Δy, and $|y| = m$.

We use these results to prove the claim in (13). The proofs of the above results are given in the Appendix. Using (15), (16) and (17) in (14) we obtain,

$$\bar{P}_2 \quad \leq \quad \sum_{i=0}^{m-1} N_i \inf_{\gamma>0}\left[F e^{\gamma n} A^i B^{m-1-i} \frac{1}{2}(1-q)^i q^{m-1-i}\right] \quad (18)$$

$$\leq \inf_{\gamma>0}\left[Fe^{\gamma n}\sum_{i=0}^{m-1}\binom{m-1}{i}((1-q)A)^i\{Bq\}^{m-1-i}\right]$$
$$\stackrel{(a)}{\leq} F\inf_{\gamma>0}\left[e^{\gamma n}\left\{(1-q)A+qB\right\}^{m-1}\right],$$

where (a) follows by using the binomial expansion of $(a+b)^{m-1}$.

As in the proof of Theorem 3.2, we use a union bound over all codewords x_2, if x_1 was the transmitted codeword to obtain,

$$\bar{P}_e \leq e^{nR}\mathbb{E}_{\mathcal{C}}[\mathbb{P}\{\text{error}|x_1, M>m\}]+\mathbb{P}\{M\leq m\} \quad (19)$$
$$\leq e^{nR}F\inf_{\gamma>0}\left[e^{\gamma n}\left\{(1-q)A+qB\right\}^{m-1}\right] \quad (20)$$
$$\leq F\inf_{\gamma>0}\left[\frac{1}{(1-q)A+qB}\left\{e^R e^{\gamma}[(1-q)A+qB]^{\frac{m}{n}}\right\}^n\right]+\delta_n$$

Therefore the probability of error goes to zero asymptotically in n if,

$$R < \sup_{\substack{\gamma>0\\0<p<1}}\left[-\theta\log\{(1-q)A+qB\}-\gamma\right] \text{ nats} \quad (21)$$

giving us the desired result.

■

Note that for $p=\frac{1}{2}$, it can be shown that the result in Theorem 3.4 reduces to that in Corollary 3.3. Also note that the optimization of (13) for a given p can be accomplished in closed form as it results in a simple quadratic equation in γ.

4 Discussion

In this paper we have developed lower bounds for the capacity of deletion channels. In order to gain insight into the behavior of deletion channels we start with some numerical examples.

In Figure 2, we plot the achievable rates derived in Sections 3.1 and 3.2. along with the upper bound derived by Ullman [10] given in (2). Note that the lower bound derived in 3.2 (labeled as the Markov bound) is non-zero up to $p_d = 0.96$ and in particular improves the previously known lower bound given in (1) [7, 11]. In Figure 3, we illustrate that for non-binary alphabet sizes, the difference between the deletion channel and the erasure channel rates can be quite small. In particular, we have compared the lower bound (6) derived in Theorem 3.2 to the erasure channel rate. As can be seen from (6) , the difference is at most $H_0(p_d) - p_d \log \frac{K}{K-1}$ bits, which is at most 1 bit. This is true with i.i.d. codebooks, and the bound becomes sharper with Markovian codebooks. This suggests that the use of sequence numbers to detect deletions is quite inefficient. For example, in TCP, 32 bits per packet are sacrificed for the sequence number, while our result shows that at most one bit of redundancy per packet is necessary to convert a deletion channel into an erasure channel.

There are several open questions that are still unresolved in this problem. Of course, the central question is the single-letter characterization of the capacity of the deletion channel. Even in the absence of such a characterization, it would be important to develop tighter upper and lower bounds for achievable rates. In particular, good upper bounds for small alphabet sizes will be useful gaining insight into the behavior of deletion channels. The problem of code construction has a long history and still has a vast number of unresolved problems (see [9] for example). If there is one message that we can give from this paper, it is that for small deletion probabilities, random i.i.d. codebooks can have good performance, but for higher deletion probabilities one needs to introduce memory into the codebook design. One way to think of Markov distributions on the codebook is to associate it with larger run-lengths in the codewords. This would also lead us to conjecture that coding in run-lengths might be a useful design technique for deletion channel codes.

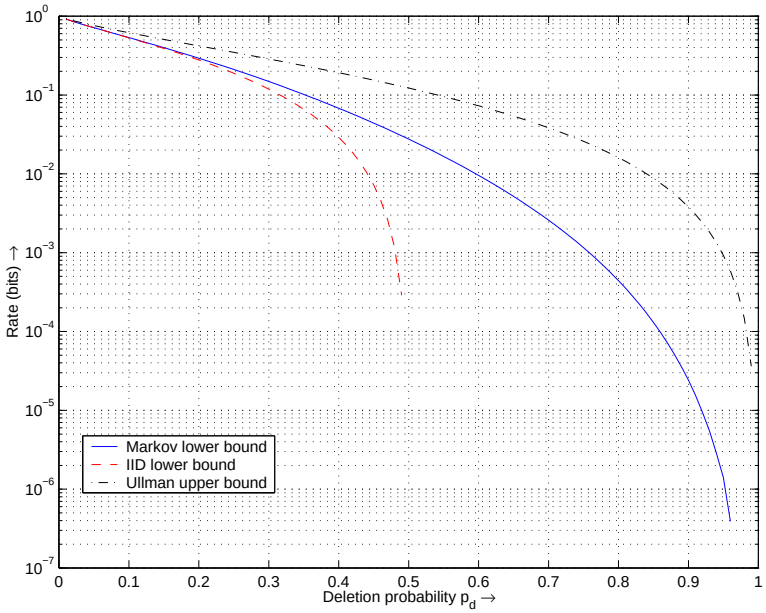

Figure 2: Achievable rates versus deletion probability for binary alphabet size.

ACKNOWLEDGMENT

We would like to thank Yiannis Kontoyiannis, Alon Orlitsky, Emre Telatar, Neil Sloane and Vinay Vaishampayan for stimulating discussions on the topic of this paper.

A Proofs of (16) and (17)

In this section we provide the details of the proof for two results used in the proof of Theorem 3.4.

A.1 Proof of (16)

In order to prove (16), we need to look at the joint process of the i.i.d. binary deletion process D and the first order Markov chain X generating the random codebook. This can be done by extending the state space of the Markov process of Figure 1 to include the state of the deletion process. This yields a four-state Markov process, (X, D), with each state being the concatenation of the state of the Markov process and the deletion process. The process $Y = D \circ X$ is generated by observing only the states where the deletion process is 0, *i.e.*, the sequence generated through the deletion channel. These states constitute a subset of the extended Markov chain state space, and by the Strong Markov Property [6], this randomly sampled Markov chain is itself a Markov chain. The transition probability $\bar{\mathbf{P}}$ of the Markov chain generated by the "watched" set is given by,

$$\bar{\mathbf{P}} = \theta \mathbf{P} \left[\mathbf{I} - (1-\theta)\mathbf{P} \right]^{-1}, \tag{22}$$

where $\mathbf{P}$ is the transition matrix of the Markov chain shown in Figure 1. It can be easily shown that the Markov chain corresponding to $\bar{\mathbf{P}}$ is also symmetric (just as the Markov chain in Figure 1) with parameter $q = 1 - \frac{1-p}{1+(1-\theta)(1-2p)}$. Therefore, the subsequence Y is obtained by running this Markov chain for m steps. This yields

$$\mathbb{P}\left\{Y \text{ with } \Delta Y \text{ of weight } i\right\} = \frac{1}{2}(1-q)^i q^{m-1-i} \tag{23}$$

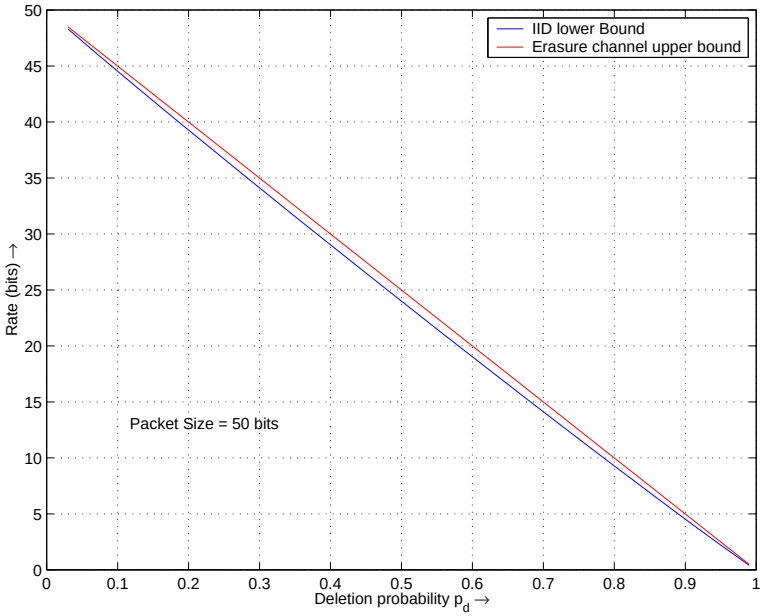

Figure 3: Erasure and deletion channels with Packet size of 50 bits.

A.2 Proof of (17)

To obtain (17), we examine the probability of a sequence y of length m occurring in a two-state Markov chain X of length n, given in Figure 1. We will consider the number of transitions N of the Markov chain needed to sequentially match every symbol of y.

As X is symmetric, the stationary probabilities for both states are identical. Thus, we can assume $y_1 = 0$ w.l.g. We first compute N_1, the number of transitions to match y_1. Its distribution is given by

$$\mathbb{P}\{N_1 = i\} = \begin{cases} \frac{1}{2} & i = 1 \\ \frac{1}{2}(1-p)p^{i-2} & i \geq 2 \end{cases}. \tag{24}$$

Now assume we have matched y_{l-1} in X, and let us consider how many transitions are needed to match y_l. We have to distinguish two cases: (a) $y_l = y_{l-1}$, and (b) $y_l \neq y_{l-1}$. Let N_a and N_b denote the number of transitions up to the next match for case (a) and (b), respectively. To this end, note that because of the symmetry of X, case (a) corresponds to the number of transitions to reach state 0 starting from state 0 (c.f. Fig. 1). For this we obtain

$$\mathbb{P}\{N_a = i\} = \begin{cases} p & i = 1 \\ (1-p)^2 p^{i-2} & i \geq 2 \end{cases}. \tag{25}$$

Similarly, case (b) corresponds to the number of transitions to reach state 1 starting from state 0. Therefore we obtain

$$\mathbb{P}\{N_b = i\} = p^{i-1}(1-p) \quad i \geq 1. \tag{26}$$

We calculate the moment generating functions for N_a and N_b as

$$\Phi_{N_a}(\gamma) \stackrel{def}{=} \mathbb{E}e^{-\gamma N_a} = (1-p)\left[\frac{e^{-\gamma}}{1-pe^{-\gamma}}\right] \tag{27}$$

where it is inherently assumed that $|pe^{-\gamma}| < 1$, which is always true because, $p \leq 1 < e^{\gamma}$ for $\gamma > 0$. Similarly, we can calculate the following,

$$\Phi_{N_b}(\gamma) \stackrel{def}{=} \mathbb{E}e^{-\gamma N_b} = (1-p)^2\left[\frac{e^{-2\gamma}}{1-pe^{-\gamma}}\right] + pe^{-\gamma}. \tag{28}$$

Let N_l denote the number of transitions to match y_l. Then

$$N_l = \begin{cases} N_1 & l = 1 \\ N_{a,l} & (\Delta y)_l = 1 \\ N_{b,l} & (\Delta y)_l = 0 \end{cases}, \tag{29}$$

where $N_{a,l}$ and $N_{b,l}$ are independent and distributed like N_a and N_b. The total number of iterations to match y is $N = \sum_{l=1}^{m} N_l$. Therefore we have

$$\mathbb{P}\{y \text{ with } \Delta y \text{ of weight } i \text{ occurs in } (X_1, \ldots, X_n)\} = \mathbb{P}\{N \leq n\} \tag{30}$$

Using the Chernoff bound we can upper bound this probability as

$$\begin{aligned} \mathbb{P}\{N \leq n\} &\leq \inf_{\gamma > 0} e^{\gamma n} \mathbb{E} e^{-\gamma \sum_{l=1}^{m} N_l} \\ &\overset{(a)}{=} \inf_{\gamma > 0} e^{\gamma n} \mathbb{E}[e^{-\gamma N_1}] \{\mathbb{E} e^{-\gamma N_a}\}^i \{\mathbb{E} e^{-\gamma N_b}\}^{m-1-i}, \end{aligned} \tag{31}$$

where (a) is obtained by the definition of N_l in (29) and the fact that the N_l are independent of each other, giving us the desired result.

References

[1] V. Chvatal and D. Sankoff. Longest common subsequence of two random sequences. *Journal of Applied Probability*, (12):306–315, 1975.

[2] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley and Sons, Inc., New York, 1991.

[3] M C. Davey and D J C. MacKay. Reliable communication over channels with insertions, deletions and substitutions. *IEEE Transactions on Information Theory*, 47(2):687–698, February 2001.

[4] Suhas N. Diggavi and Matthias Grossglauser. Information transmission over a finite buffer channel. In *IEEE International Symposium on Information Theory (ISIT)*, page 52, June 2000.

[5] R L. Dobrushin. Shannon's theorems for channels with synchronization errors. *Problems Information Transmission*, 3(4):11–26, 1967. Translated from Problemy Peredachi Informatsii, vol. 3, no. 4, pp 18–36, 1967.

[6] Richard Durrett. *Probability: theory and examples*. Duxbery Press, 2nd edition, 1995.

[7] R G. Gallager. Sequential decoding for binary channels with noise and synchronization errors. Lincoln Lab. Group Report, October 1961.

[8] V I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics - Doklady*, 10(8):707–710, February 1966.

[9] N J A. Sloane. On Single-Deletion-Correcting Codes. To appear in D. Ray-Chaudhuri Festschrift, 2001. See: http://www.research.att.com/ njas/doc/dijen.ps.

[10] Jeffrey D. Ullman. On the capabilities of codes to correct synchronization errors. *IEEE Transactions on Information Theory*, 13(1):95–105, January 1967.

[11] K Sh. Zigangirov. Sequential decoding for a binary channel with drop-outs and insertions. *Problems Information Transmission*, 5(2):17–22, 1969. Translated from Problemy Peredachi Informatsii, vol. 5, no. 2, pp 23–30, 1969.