

Balancing forward and feedback error correction for erasure channels with unreliable feedback

Anant Sahai
sahai@eecs.berkeley.edu

Abstract

The traditional information theoretic approach to studying feedback is to consider ideal instantaneous high-rate feedback of the channel outputs to the encoder. This was acceptable in classical work because the results were negative: Shannon pointed out that even perfect feedback often does not improve capacity and in the context of symmetric DMCs, Dobrushin showed that it does not improve the fixed block-coding error exponents in the interesting high rate regime. However, it has recently been shown that perfect feedback **does** allow great improvements in the asymptotic tradeoff between end-to-end delay and probability of error, even for symmetric channels at high rate. Since gains are claimed with ideal instantaneous feedback, it is natural to wonder whether these improvements remain if the feedback is unreliable or otherwise limited.

Here, packet-erasure channels are considered on both the forward and feedback links. First, the feedback channel is considered as a given and a strategy is given to balance forward and feedback error correction in the suitable information-theoretic limit of long end-to-end delays. At high enough rates, perfect-feedback performance is asymptotically attainable despite having only unreliable feedback! Second, the results are interpreted in the zero-sum case of “half-duplex” nodes where the allocation of bandwidth or time to the feedback channel comes at the direct expense of the forward channel. It turns out that even here, feedback is worthwhile since dramatically lower asymptotic delays are possible by appropriately balancing forward and feedback error correction.

The results easily generalize to channels with strictly positive zero-undeclared-error capacities.

Index Terms

Feedback, hybrid ARQ, delay, reliability functions, random coding

Balancing forward and feedback error correction for erasure channels with unreliable feedback

I. INTRODUCTION

The three most fundamental parameters when it comes to reliable data transport are probability of error, end-to-end system delay, and data rate. Error probability is critical because a low probability of bit error lies at the heart of the digital revolution justified by the source/channel separation theorem. Delay is important because it is the most basic cost that a system must pay in exchange for reliability — it allows the laws of large numbers to be harnessed to smooth out the variability introduced by random communication channels.

Data rate, while apparently the most self-evidently important of the three, also brings out the seemingly pedantic question of units: do we mean bits per second or bits per channel use? In a point-to-point setting without feedback, there is an unambiguous mapping between the two of them given by the number of channel uses per second. When feedback is allowed, an ambiguity arises: how should the feedback channel uses be accounted for? Are they “free” or do they need to be counted?

Traditionally, the approach has been to consider feedback as “free” because the classical results showed that in many cases even free feedback does not improve the capacity [1], and in the fixed-length block-coding context, does not even improve the tradeoff between the probability of error and delay in the high-rate regime for symmetric channels [2], [3]. (See [4] for a detailed review of this literature.) Thus, the answer was simple: if feedback comes at any cost, it is not worth using for memoryless channels.

In [4], we recently showed that perfect feedback is indeed quite useful in a “streaming” context if we are willing to use non-block codes to implement our communication system. In the natural setting of a message stream being produced at a fixed (deterministic) rate of R bits per second, feedback *does* provide¹ a tremendous advantage in terms of the tradeoff between end-to-end delay and the probability of bit error. As the desired probability of error tends to zero, feedback reduces the end-to-end delay by a *factor* that approaches infinity as the desired rate approaches capacity. In [4], the resulting fixed-delay error exponents with feedback are calculated exactly for erasure channels and channels with strictly positive zero-error capacity. For general DMCs, [4] gives a general upper bound (the uncertainty-focusing bound) along with a suboptimal construction that substantially outperforms codes without feedback at high rates.

Once it is known that perfect feedback is very useful, it is natural to ask whether this advantage persists if feedback is costly, rate-limited, or unreliable in some way. After all, the real question is not whether perfect feedback would be useful but how imperfect feedback is worth designing in real communication systems. This has long been recognized as the Achilles’ Heel for the information-theoretic study of feedback. Bob Lucky in [6] stated it dramatically:

Feedback communications was an area of intense activity in 1968... A number of authors had shown constructive, even simple, schemes using noiseless feedback to achieve Shannon-like behavior... The situation in 1973 is dramatically different... *The subject itself seems to be a burned out case.* ...

In extending the simple noiseless feedback model to allow for more realistic situations, such as noisy feedback channels, bandlimited channels, and peak power constraints, theorists discovered a certain “brittleness” or sensitivity in their previous results.

The literature on imperfect feedback in the context of memoryless channels is relatively thin. Schulman and others have studied interaction over noisy channels in the context of distributed computation [7], [8]. The computational agents can only communicate with each other through noisy channels in both directions. In Schulman’s formulation, neither capacity nor end-to-end delay is a question of major interest since constant factor slowdowns are seemingly unavoidable. The focus is instead on making sure that the computation succeeds and that the slowdown factor does not scale with problem size (as it would for a purely repetition based strategy).

On the reliability side before recently, all the limited successes were for continuous-alphabet AWGN type-channels following the Schalkwijk/Kailath model from [9], [10]. Kashyap in [11] introduced a scheme that tolerated noise on the feedback link, but it used asymptotically infinite (in the block-length) power on the feedback link to overcome

¹This overturned Pinsker’s incorrect assertion in Theorem 8 of [5] that feedback gives no asymptotic advantage in this nonblock setting.

it. It is only the work by Kramer [12] and Lavenburg [13], [14] that worked with finite average power in both directions. But these were all cases in which the average nature of the power-constraint played an important role. Recently, the AWGN story with noisy feedback has also attracted the interest of Weissman, Lapidot and Kim in [15], who rigorously proved a strongly negative folk result for the case of *uncoded* channel-output feedback corrupted by arbitrarily low levels of Gaussian noise. At the same time, [16] showed that if the feedback noise has bounded support, then techniques similar to those of [17] could preserve reliability gains, but only at the price of having to back away from the capacity of the forward link.

For finite alphabets, we have recently had some success in showing robustness to imperfect feedback in the “soft deadlines” context where the decoder is implicitly allowed to postpone making a decision, as long as it does not do so too often. With perfect feedback, this has traditionally been considered in the variable-block-length setting where Burnashev’s bound of [18] gives the ultimate limit with perfect feedback and Yamamoto-Itoh’s scheme of [19] provides the baseline architecture. [20] showed that if the feedback channel was noisy, but of very high quality, then the loss relative to the Burnashev bound could be quite small by appropriately using anytime codes and pipelining. [21] allowed bursty noiseless feedback, but constrained its overall rate to show that by using hashing ideas, something less than full channel output feedback could be used while still attaining the Burnashev bound. The ideas of [20], [21] were combined in [22] to show that it was possible to get reliability within a factor of two of the Burnashev bound as long as the noisy feedback channel’s capacity was higher than the capacity of the forward link and this was further tightened in [23] to a factor that approaches one as the target rate approached capacity. This story culminates in [24] where it is shown that from the system-level perspective, Burnashev’s bound is not the relevant target. Instead, Kudryashov’s performance with noiseless feedback in [25] (better than the Burnashev Bound) can in fact be asymptotically attained robustly as long as the feedback channel’s capacity is larger than the target reliability.

The focus here is on the problem of *fixed* rate and *fixed* end-to-end delay in the style of [4] where the decoder is not allowed to postpone making a decision. This paper restricts attention to the case of memoryless packet-erasure channels where the feedback path is also an unreliable packet-erasure channel. Recently, Massey [26] has had some interesting thoughts on this problem, but he claims no asymptotic reliability gains for uncertain feedback over no feedback. The issue of balancing forward and feedback error correction has also attracted attention in the networking community (see eg [27]), but the focus there is not purely on reliability or delay but is mixed with the issue of adapting to bursty channel variation as well as fairness with other streams.

Section II establishes the notation and states the main results, with the proofs following in subsequent sections. The results are stated for the concrete case of packet-erasure channels. Section II also plots the performance for some examples and compares the results with the baseline approaches of only using forward error correction and just using feedback for requesting retransmissions at the individual packet level.

In Section III, the feedback channel uses are considered “free” in that they do not compete with forward channel uses for access to the underlying communication medium. Adapting arguments from [4], it is shown that asymptotically, perfect feedback performance can be attained even with unreliable feedback. Because the uncertainty-focusing bound of [4] is met at rates that are high enough, it is known that this is essentially optimal. If the target rate is too low, then the dominant error event for the scheme turns out to be the feedback channel going into complete outage and erasing every packet.

At the end of Section III, it is noted that the main result generalizes naturally from packet-erasure channels to DMCs whose zero-undetected-error capacity is equal to their Shannon capacity. As a bonus², the same techniques give rise to a generalization of Theorem 3.4 of [4] and show the achievability with perfect feedback of the symmetric uncertainty-focusing bound at high rate for any channel whose probability matrix contains a nontrivial zero.

Section IV reinterprets the results of the previous section to address the question of “costly” feedback in that both the data rate and delay are measured not relative to forward channel uses, but relative to the sum of feedback and forward channel uses. This models the case when the feedback channel uses the same underlying communication resource as the forward channel in a zero-sum way (e.g. time-division or frequency-division in wireless networks). It is shown that there is a tremendous advantage to using some of the channel uses for feedback.

²Dear Reviewers: the result referred to here does not really fit in with the overall theme of unreliable feedback, but is placed in this paper since the techniques used are common.

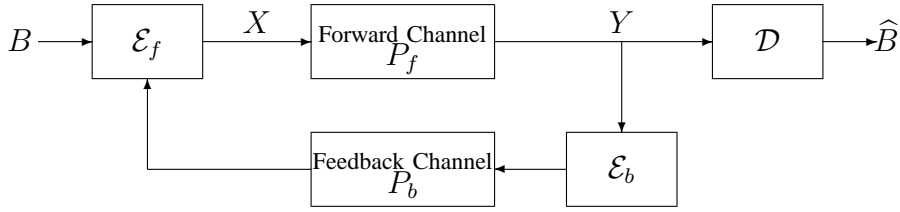


Fig. 1. The problem of communication with unreliable feedback. For the most part in this paper, both of the unreliable channels P_f and P_b are packet erasure channels that are independent of each other.

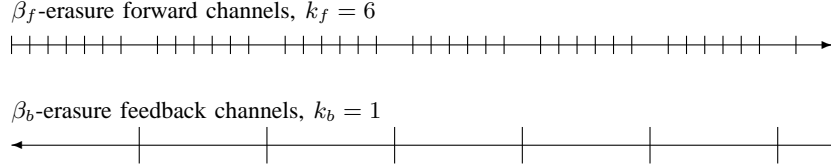


Fig. 2. Feedback and forward channel uses. The forward channel has packet-erasure probability β_f while the feedback channel has packet-erasure probability β_b . In this picture $k_f = 6, k_b = 1$ So a fraction $\frac{1}{7}$ of the total channel uses are allocated to feedback.

II. DEFINITIONS AND MAIN RESULTS

The problem and notation is illustrated in Figures 1 and 2. Formally:

Definition 2.1: A C_p -bit β -erasure channel refers to a discrete memoryless channel that accepts packets $X(t)$ consisting of C_p bits (thought of as integers from 0 to $2^{C_p} - 1$ or strings from $\{0, 1\}^{C_p}$) per packet as inputs and either delivers the entire packet perfectly $Y(t) = X(t)$ with probability $1 - \beta$ or erases the whole packet $Y(t) = -1$ with probability β .

Definition 2.2: The $(k_f, k_b, C_f, C_b, \beta_f, \beta_b)$ problem consists of a system in which one cycle of interaction between encoders and decoders consists of k_f independent packets being sent along a C_f -bit β_f -erasure channel along the forward direction and k_b independent packets being sent along a C_b -bit β_b -erasure channel along the reverse direction.

A *feedback encoder* \mathcal{E}_b for this problem is a sequence of maps $\mathcal{E}_{b,t}$. The range of each map is k_b packets $(X_b(k_b t + 1), X_b(k_b t + 2), \dots, X_b(k_b t + k_b))$ consisting of C_b bits each. The t -th map takes as input all the available forward channel outputs $(Y_f(1), \dots, Y_f(k_f t))$ so far.

A *rate R forward encoder* \mathcal{E}_f for this problem is also a sequence of maps $\mathcal{E}_{f,t}$. The range of each map is k_f packets $(X_f(k_f t + 1), X_f(k_f t + 2), \dots, X_f(k_f t + k_f))$ consisting of C_f bits each. The t -th map takes as input all the available feedback channel outputs $(Y_b(1), \dots, Y_b(k_b t))$ as well as the message bits $B_1^{\lfloor (Rk_f C_f)t \rfloor}$ so far.

The rate R above is in terms of forward channel uses only and is normalized in units of C_f -bit packets. The rate R' in terms of overall channel uses is $R' = R \frac{k_f}{k_f + k_b}$. The rate \bar{R} in terms of weighted channel uses is $\bar{R} = R \frac{k_f C_f}{k_f C_f + k_b C_b}$.

A *delay d rate R decoder* is a sequence of maps \mathcal{D}_i . The range of each map is an estimate \hat{B}_i for the i -th bit taken from $\{0, 1\}$. The i -th map takes as input the available channel outputs $(Y_f(1), Y_f(2), \dots, Y_f(\lceil \frac{i}{Rk_f C_f} \rceil + dk_f))$. This means that it can see dk_f channel uses beyond when the bit to be estimated first had the potential to influence the channel inputs.

Just as rate can be expressed in different units, so can delay. The delay d above is in terms of forward channel uses only. The delay d' in terms of overall channel uses is $d' = d \frac{k_f + k_b}{k_f}$. The delay \bar{d} in terms of weighted channel uses is $\bar{d} = d \frac{k_f C_f + k_b C_b}{k_f C_f}$.

All encoders and decoders are assumed to be randomized and have access to infinite amounts of common randomness that is independent of both the messages as well as the channel unreliability.

The notation above captures the real flexibility of interest. The distinction between C_f and C_b allows for forward and feedback packets to be of different size. The distinction between k_f and k_b summarizes the relative width of

the forward and feedback pipes. Similarly, the distinction between β_f and β_b allows for the channel to be more or less reliable in the different directions.

The three kinds of units correspond to three different ways of thinking about the problem.

- The unadorned R and d take the traditional approach of considering the feedback to be entirely free, although there is now a limited amount of it and it is imperfect.
- The R' and d' consider feedback to be costly, but in terms of channel uses only. The relative size of the packets is considered unimportant. It is clear that as long as some feedback is present, $R' < R$ and $d' > d$. These metrics give an incentive to use less feedback if possible.
- The \bar{R} and \bar{d} also consider the relative size of the packets significant. These metrics give an incentive to use shorter feedback packets.

It is even interesting to consider combinations of metrics. The combination of \bar{R} and d' is particularly interesting since it corresponds to the case when feedback is implemented as bits stolen from message-carrying packets coming in the reverse direction. Using \bar{R} gives an incentive to make the number of bits taken small, but using d' captures the fact that the delay in real time units includes the full lengths of both the intervening forward and feedback packets.

Definition 2.3: The fixed-delay error exponent α is asymptotically *achievable* at message rate R across a noisy channel if for every delay d_j in some increasing sequence $d_j \rightarrow \infty$ there exist rate R encoders $(\mathcal{E}_f^j, \mathcal{E}_b^j)$ and delay d_j decoders \mathcal{D}^j that satisfy the following properties when used with input bits B_i drawn from iid fair coin tosses.

- 1) For the j -th code, there exists an $\epsilon_j < 1$ so that $\mathcal{P}(B_i \neq \hat{B}_i(d_j)) \leq \epsilon_j$ for every bit position $i \geq 1$. The $\hat{B}_i(d_j)$ represents the delay d_j estimate of B_i produced by the $(\mathcal{E}_f^j, \mathcal{E}_b^j, \mathcal{D}^j)$ triple connected through the channels in question.
- 2) $\lim_{j \rightarrow \infty} \frac{-\ln \epsilon_j}{d_j k_f} \geq \alpha$.

The exponent α is asymptotically *achievable universally over delay* or in an *anytime fashion* if a single encoder pair $(\mathcal{E}_f^j, \mathcal{E}_b^j)$ can be used simultaneously for all sufficiently long d_j . Only the decoders \mathcal{D}^j have to change with the target delay d_j .

The error exponents α' and $\bar{\alpha}$ are defined analogously but use the d'_j and \bar{d}_j versions of delay.

A. Main Results

Theorem 2.1: Given the $(k_f, k_b, C_f, C_b, \beta_f, \beta_b)$ problem with $k_f, k_b \geq 1$, forward packet size $C_f \geq 1$, and feedback packet size $C_b \geq 1$, it is possible to asymptotically achieve all fixed-delay reliabilities

$$\alpha < \min \left(-\frac{k_b}{k_f} \ln \beta_b, E_0(C_f, 1) \right) \quad (1)$$

where the Gallager function for the forward channel is

$$E_0(C_f, \rho) = -\ln(\beta_f + 2^{-\rho C_f}(1 - \beta_f)), \quad (2)$$

as long as the rate R in normalized C_f units satisfies

$$R < \frac{\alpha}{\alpha + \ln \left(\frac{1 - \beta_f}{1 - \exp(\alpha) \beta_f} \right)}. \quad (3)$$

Furthermore, these fixed-delay reliabilities are obtained in an anytime fashion.

Proof: See Section III-A.

The uncertainty-focusing bound from [4] for this problem assuming *perfect feedback* is easily calculated to be given by (3) but it holds for all $0 < \alpha < -\ln \beta_f$. Since lower reliabilities are associated with higher rates, this shows that the result of Theorem 2.1 is asymptotically optimal at high enough rates. The feedback packet size needs to be only one bit and there is similarly no restriction on the size of forward packets. Since $\lim_{C_f \rightarrow \infty} E_0(C_f, 1) = -\ln \beta_f$, the sense of high enough rates given by (1) depends only on the relative frequency and reliability of the feedback link as the forward packet size tends to infinity.

When the packet sizes are at least two bits long in both directions, asymptotic delay performance can be slightly improved at low rates.

Theorem 2.2: Given the $(k_f, k_b, C_f, C_b, \beta_f, \beta_b)$ problem with $k_f, k_b \geq 1$, $C_f \geq 2$, and feedback packet size $C_b \geq 2$, it is possible to asymptotically achieve all fixed-delay reliabilities

$$\alpha < -\frac{k_b}{k_f} \ln \beta_b \quad (4)$$

as long as the rate R in normalized C_f units satisfies

$$R < \left(\frac{C_f - 1}{C_f} \right) \frac{\alpha}{\alpha + \ln \left(\frac{1 - \beta_f}{1 - \exp(\alpha)\beta_f} \right)}. \quad (5)$$

Furthermore, these fixed-delay reliabilities are obtained in an anytime fashion.

Proof: See Section III-B.

The upper-limit on reliability for the scheme given by (4) corresponds to the event that the feedback channel erases every feedback packet for the entire duration of d cycles. If k_f and k_b could be chosen by the system designer, this constraint could be made non-binding simply by choosing $\frac{k_b}{k_f}$ large enough. However, if there is such flexibility, it is only fair to also penalize based on the total resources used, rather than only penalizing forward channel uses.

Theorem 2.3: Given only $(C_f \geq 1, C_b \geq 1, \beta_f > 0, \beta_b > 0)$, the $k_f > 0$ and $k_b > 0$ can be chosen to asymptotically achieve all (R', α') pairs that are contained within the parametric region:

$$\begin{aligned} \alpha' &< E'_0(C_f, \rho), \\ R' &< \frac{E'_0(C_f, \rho)}{\rho C_f \ln 2} \end{aligned} \quad (6)$$

where

$$E'_0(C_f, \rho) = ((-\ln \beta_f)^{-1} + (E_0(C_f, \rho))^{-1})^{-1} \quad (7)$$

and the Gallager function $E_0(C_f, \rho)$ is defined in (2) and ρ ranges from 0 to 1.

If furthermore $C_f \geq 2, C_b \geq 2$, then the following region is also attainable:

$$\begin{aligned} \alpha' &< E'_0(C_f - 1, \rho), \\ R' &< \frac{E'_0(C_f - 1, \rho)}{\rho C_f \ln 2} \end{aligned} \quad (8)$$

with ρ ranging from 0 to ∞ .

If C_b stays constant while C_f can be chosen as large as desired, then (\bar{R}, α') in

$$\begin{aligned} \alpha' &< E'_0(C_f, \rho), \\ \bar{R} &< \frac{E_0(C_f, \rho)}{\rho C_f \ln 2} \end{aligned} \quad (9)$$

can be achieved where $\rho \in [0, 1]$. If the $(\bar{R}, \bar{\alpha})$ tradeoff is desired, use (9) for \bar{R} with $\bar{\alpha} < E_0(C_f, \rho)$.

All of these fixed-delay reliabilities are obtained in an anytime fashion.

Proof: See Section IV.

B. Pure strategies and comparison plots

To understand the implications of Theorems 2.1, 2.2 and 2.3, it is useful to compare them to what would be obtained using strategies for reliable communication that use only feedback error correction or only forward error correction.

1) *Pure forward error correction*: The simplest approach is to ignore the feedback and just use a fixed-delay code. [28], [29] reveal that infinite-constraint-length convolutional codes achieve the random-coding error exponent $E_r(R)$ with respect to end-to-end delay and [4] tells us that we cannot do any better than the sphere-packing bound without feedback.

$$E_r(R) = \max_{\rho \in [0,1]} E_0(C_f, \rho) - \rho R C_f \quad (10)$$

$$E_{sp}(R) = \max_{\rho \in [0,\infty)} E_0(C_f, \rho) - \rho R C_f \quad (11)$$

where R ranges from 0 to the forward capacity of $1 - \beta_f$ packets (of size C_f each) per channel use.

Using a fixed-length block code would introduce another factor of two in end-to-end delay since the message to be transmitted would first have to be buffered up to make a block.

2) *Pure feedback error correction*: The intuitive “repeat until success” strategy for perfect feedback analyzed in [4] can be adapted to when the feedback is unreliable. For simplicity, focus on $k_f = k_b = 1$. The idea is for the feedback encoder to use 1 bit on the feedback channel to indicate if the forward packet was received or not. If this feedback packet is not erased, the situation is exactly as it is when feedback is perfect. When the feedback packet is erased, the safe choice for the forward encoder is to retransmit. However, this requires some way for the decoder to know that the incoming packet is a retransmission rather than a new packet. The practical way this problem is solved is by having sequence numbers on packets. As [26] points out, only 1 bit of overhead per forward packet is required for the sequence number in this scenario.

Thus, the resulting system behaves like a repeat until success system with perfect feedback with two modifications:

- The effective forward packet size goes from C_f bits to $C_f - 1$ bits to accommodate the 1-bit sequence numbers.
- The effective erasure probability goes from β_f to $(1 - (1 - \beta_f)(1 - \beta_b))$ because an erasure on either forward or feedback channel demands a retransmission.

Thus an error exponent of α with respect to end-to-end delay can be achieved as long as the rate R (in units of C_f bits at a time) satisfies (using Theorem 3.3 in [4] and adjusting for α being in base e rather than base 2):

$$R < \left(\frac{C_f - 1}{C_f} \right) \frac{\alpha}{\alpha + \ln \left(\frac{(1 - \beta_f)(1 - \beta_b)}{1 - \exp(\alpha)(1 - (1 - \beta_f)(1 - \beta_b))} \right)} \quad (12)$$

and $0 < \alpha < -\ln(1 - \beta_f)(1 - \beta_b)$. No higher α can be achieved by this scheme. Even as $\alpha \rightarrow 0$ and $C_f \rightarrow \infty$, the above rate only reaches $(1 - \beta_f)(1 - \beta_b)$ and thus is bounded away from the capacity $1 - \beta_f$ of the forward link.

3) *Comparison*: Three scenarios are considered. Figure 3 sets $k_f = k_b = 1$ and compares the pure feedback and pure forward error correction strategies to the balanced approach of Theorem 2.1 when the erasure probability on both the forward and feedback links are the same. The limit of $C_f \rightarrow \infty$ is shown, although this is only significant at low rates. If the feedback link were more unreliable than the forward link, then the reliability gains from Theorem 2.1 would saturate at lower rates. Looking at the curves in the vicinity of capacity shows clearly that the factor reduction in asymptotically required end-to-end delay over purely forward error correction tends to ∞ .

Figure 4 illustrates the difference between Theorems 2.1 and 2.2. For high rates, Theorem 2.1 is better. But at low rates, Theorem 2.2 provides better reliability and hence shorter asymptotic end-to-end delays. When the packets are short, the capacity penalty for allocating 1 bit for a header can be significant as the plot illustrates using $C_f = 4$.

Figure 5 illustrates the scenario of Theorem 2.3 in that it assumes that there is a single shared physical channel that must be divided between forward and feedback channel uses. Somewhat surprisingly, feedback becomes more valuable the closer the system comes to capacity. The factor reduction in asymptotically required end-to-end delay over purely forward error correction tends to ∞ as the data rate approaches capacity. This shows that, at least in the packet-erasure case, feedback is worth implementing even if it comes at the cost of taking resources away from the forward path.

Finally, Figure 6 illustrates the impact of how rate and delay are counted. Notice that at high rates, the curve in which the feedback is counted against the delay but not the rate is very close to the case in which feedback is free. This makes sense when the packets are large and there is presumably independent data coming in the opposite direction.

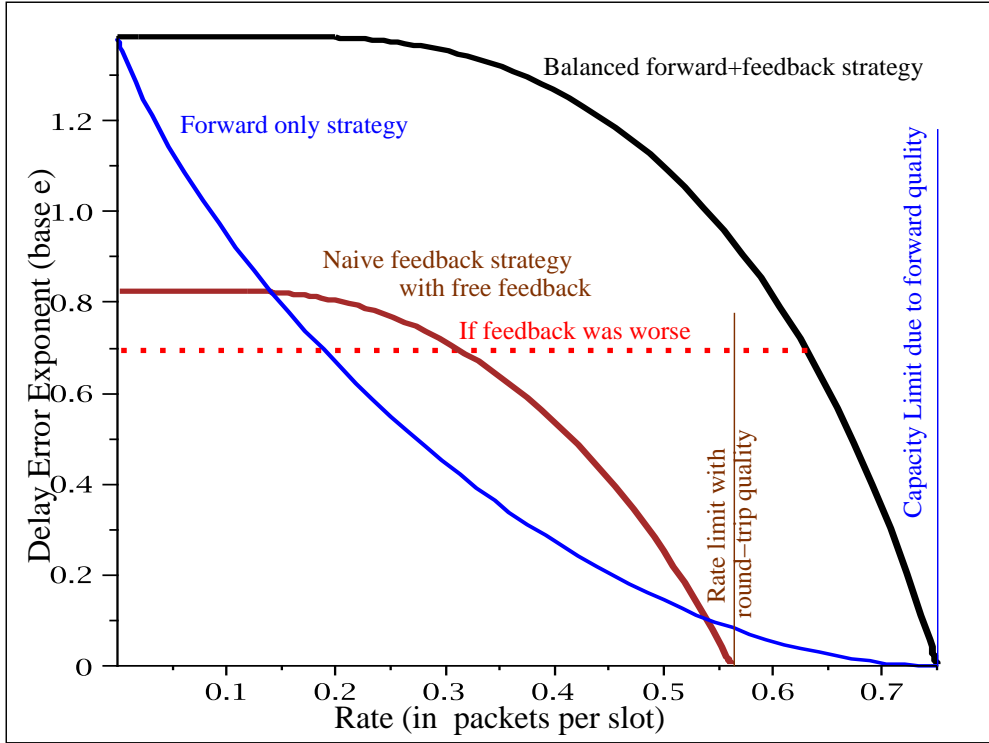


Fig. 3. The error exponents governing the asymptotic tradeoff between the probability of error and end-to-end delay for an 0.25-erasure channel with a separate 0.25-erasure channel on the feedback link. The top curve is the *uncertainty-focusing bound* from [4] that is optimal assuming that feedback is perfect. If the probability of erasure on the reverse channel were increased to 0.50, the achievable exponents for the schemes of this paper saturate at $\ln 0.5$. For comparison, the simple feedback-only (with 0.25 erasure on both forward and feedback links) strategy achieves only the lower curve. The forward-only curve bounds what is possible without using feedback in general and also what is possible with feedback if the system is restricted to fixed-length block codes.

III. UNRELIABLE, BUT “FREE” FEEDBACK

Throughout this section, the k_f and k_b are considered to be a given. The goal is to prove Theorems 2.1 and 2.2. The basic idea is to adapt the (n, c, l) scheme of [4] to this situation. Corollary 6.1 of [4] is the key tool used to prove the results.

A. Theorem 2.1: no list decoding

The scheme used is:

- 1) Group incoming bits into blocks of size nck_fRC_f each. Assume that n and c are both large, but nck_f is small relative to the target end-to-end delay measured in forward channel uses.
- 2) Hold blocks in a FIFO queue awaiting transmission. The first block is numbered 1 with numbers incrementing by 1 thereafter. At time 0, both sides agree that block 0 has been decoded. The current pointer for both is set at 1.
- 3) The forward encoder transmits the oldest waiting block using an ∞ -length random codebook (rateless code) with a new codebook being drawn for each block. The codewords themselves consist of iid uniform C_f -bit packets.

Formally, the codewords are $X_i(j, t)$ where $i > 0$ represents the current block number, $t > 0$ is the current time, and $0 \leq j < 2^{nck_fRC_f}$ is the value of the current block. Each $X_i(j, t)$ is drawn iid and is k_f packets long.

- 4) The forward decoder uses the received channel symbols $Y(t)$ to eliminate potential messages (codewords) that could have been sent as the current block i . As soon as there is only one solitary codeword j left, the decoder considers it to be the true value \hat{j} for that block and the block is marked as successfully decoded.

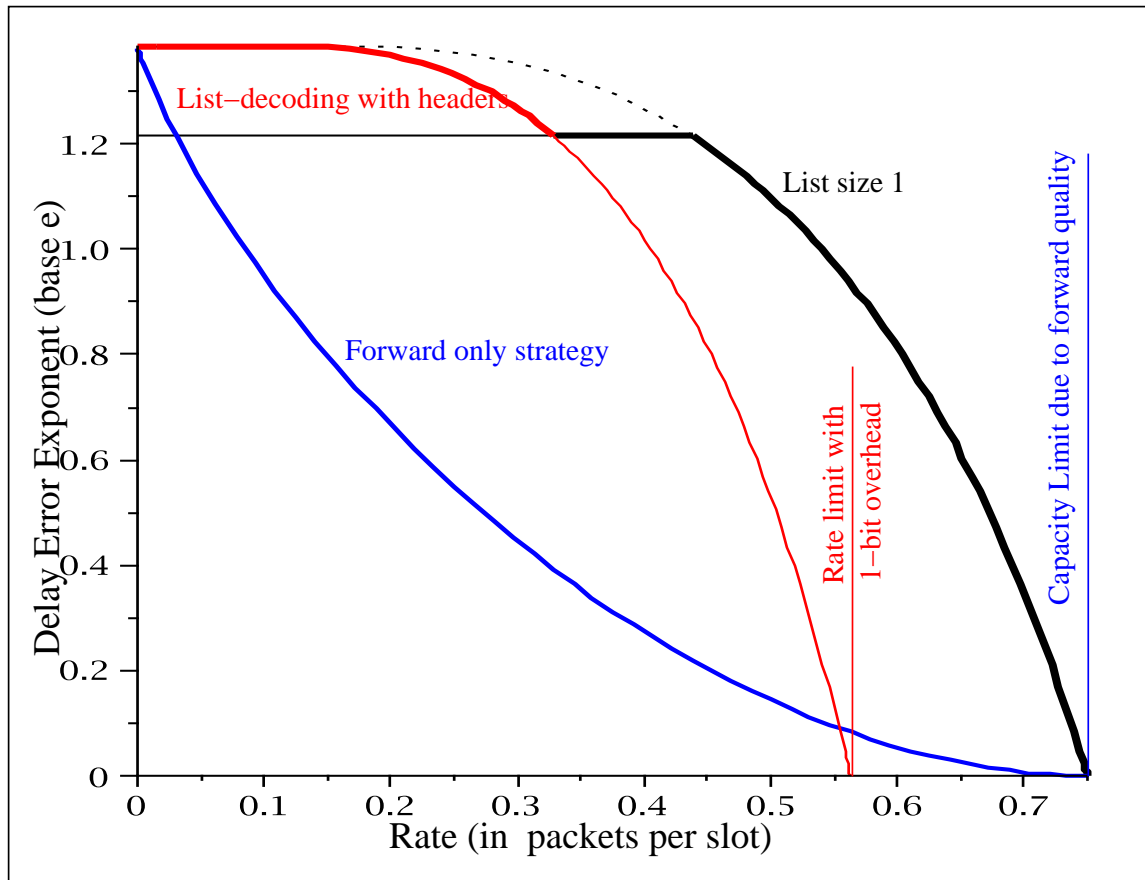


Fig. 4. The impact of list decoding is significant only at low rates. The top curve is the uncertainty-focusing bound. The dotted segment is the part that is not achieved. The red curve represents what is achievable using Theorem 2.2 and the thick black curve represents Theorem 2.1. The discontinuity in the thick curve indicates when it is worth switching between the two theorems. The forward packet size $C_f = 4$ and the probability of erasure is 0.25 on both the forward and feedback links.

When the current received packet $Y(t)$ is incompatible with this solitary codeword (ie $-1 \neq Y(t) \neq X_i(\hat{j}, t)$), then the current block count i is incremented at the decoder and it considers the next block to have begun.

- 5) The feedback encoder always uses its one bit to send back the modulo 2 number of the last block (usually $i - 1$, but sometimes i when the current block has been decoded and the receiver is still waiting for a sign that the next block has begun) that was successfully decoded.
- 6) If the forward encoder receives feedback indicating that the current block has been successfully decoded, it removes the current block from the queue, increments the current i pointer, and moves on to the next block. If there are no blocks awaiting transmission, the encoder can just continue extending the current codeword until there is something new to send.

An interesting feature of this scheme is that *there are no explicit sequence numbers* on the forward packets unlike the approach of [26]. Instead, they are implicit. This prevents a loss of rate. Synchronization between the forward encoder side and decoder side is maintained because:

- They start out synchronized.
- The forward encoder can only increment its pointer after getting explicit feedback from the decoder telling it that the block has been correctly received. Because the feedback channel is an erasure channel, this implies that such an acknowledgement was actually sent.
- The decoder can only increment its pointer i after receiving a symbol that is incompatible with the prior codeword. Because the forward channel is an erasure channel, the only way this can happen is if the packet was indeed sent from a new codebook indicating unambiguously that the forward encoder has incremented its pointer.

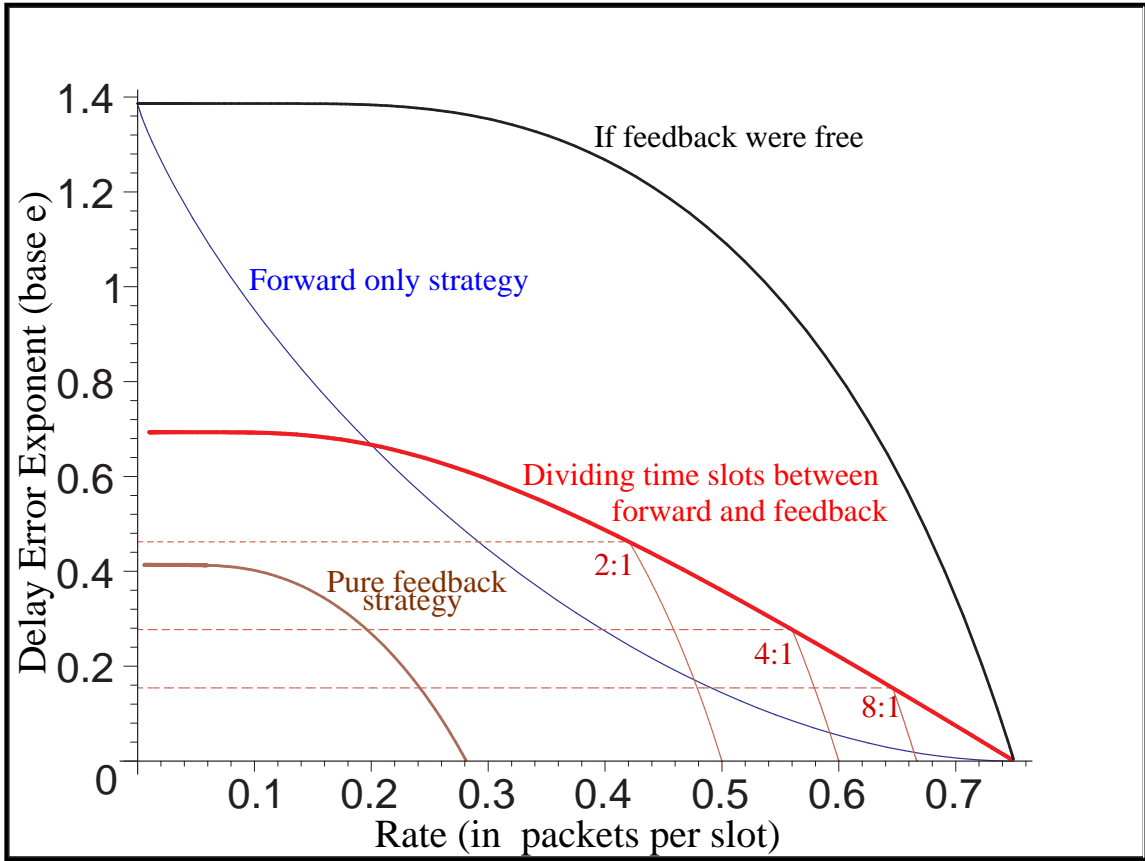


Fig. 5. The upper curve is the uncertainty-focusing bound with perfect and free feedback and the lower-most curve is the delay performance attained by feedback-only error-correction strategies that split the channel equally among forward and feedback uses. The intermediate curves reflect giving everything to the forward channel and various ratios of forward to feedback channel uses. The envelope of such schemes (described in Theorem 2.3) is also plotted.

It is thus clear that no errors are ever made. The total delay experienced by a bit can be broken down into three parts:

- 1) *Assembly delay*: How long it takes before the rest of the message block has arrived at the forward encoder. (Bounded by the constant nck_f forward channel uses and hence asymptotically irrelevant as $d \rightarrow \infty$.)
- 2) *Queuing delay*: How long the message block must wait before it begins to be transmitted.
- 3) *Transmission delay*: How many channel uses it takes before the codeword is correctly decoded (Random quantity T'_i that must be an integer multiple of k_f .)

To understand this, a closer look at the transmission delay T'_i is required. First, the T'_i can be upper-bounded by the service time T_i that measures how long it takes till the forward encoder is sure that the codeword was correctly decoded. This puts us in the setting of Corollary 6.1 of [4].

1) *The service time*: $T_i = T_{1,i} + T_{2,i} + T_{3,i}$ consists of the sum of how long it takes to complete three distinct stages of service.

- $T_{1,i}$: How long till the decoder realizes that the forward encoder has moved on.

Since the new codeword's symbol $X_i(j, t)$ is drawn independently from the previous codeword's symbol $X_{i-1}(j_{prev}, t)$, the probability of a received packet being ambiguous is just $\beta_f + (1 - \beta_f)2^{-C_f}$ since there is a β_f probability of being erased and only a 1 in 2^{C_f} chance of drawing something identical. Thus:

$$\begin{aligned}
 \mathcal{P}(T_{1,i} > t) &= (\beta_f + (1 - \beta_f)2^{-C_f})^t \\
 &= \exp(t \ln(\beta_f + (1 - \beta_f)2^{-C_f})) \\
 &= \exp(-tE_0(C_f, 1))
 \end{aligned}$$

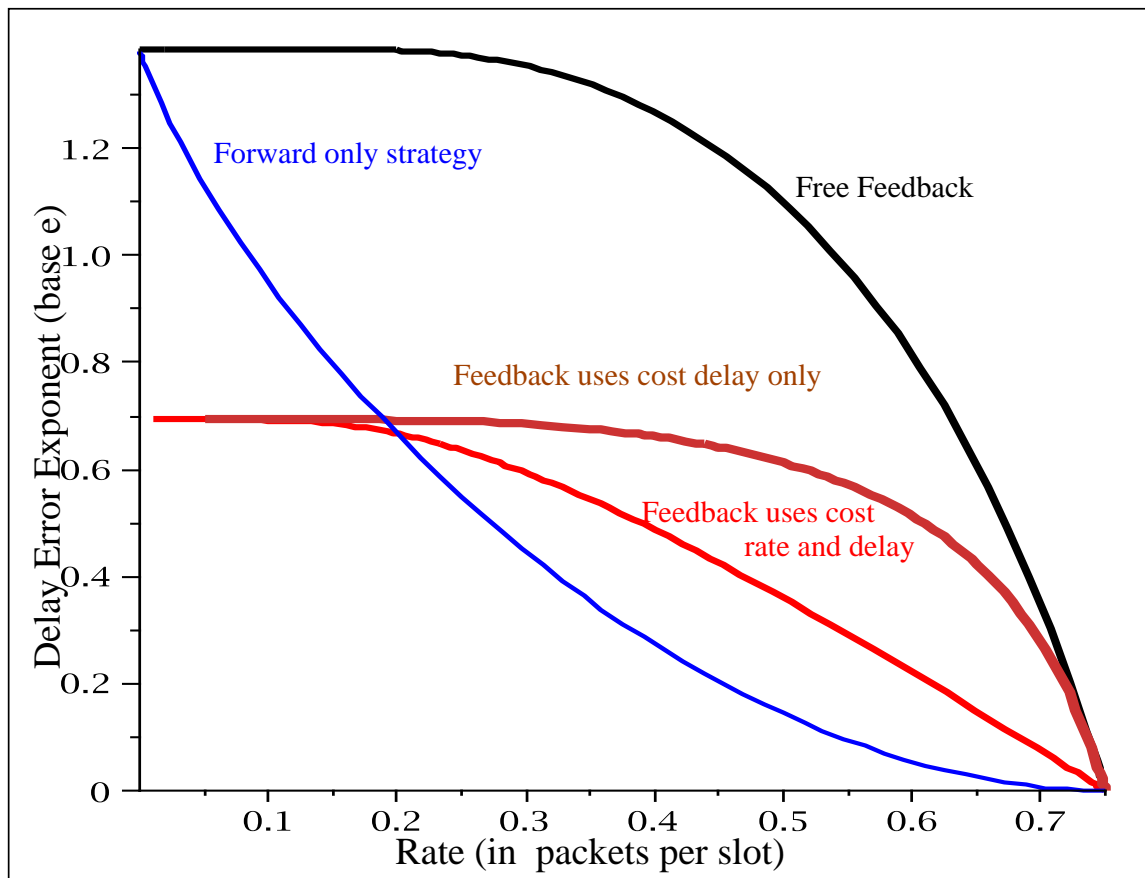


Fig. 6. The upper curve is the uncertainty-focusing bound with perfect and free feedback. The second curve down optimizes the split between forward and feedback channel uses and counts end-to-end delay in terms of total channel uses. Rate is calculated only relative to the forward channel uses with the idea that the feedback packets are carrying other useful data. The third curve is the one from Figure 5 that counts both delay and rate relative to total channel uses. The final curve is for forward error-correction only.

and so $T_{1,i}$ has a geometric distribution governed by the exponent $E_0(C_f, 1)$. (Alternatively, this can be seen directly from the interpretation of $E_0(1)$ as the exponent governing the pairwise probability of confusion for codewords for the regular union bound [30].)

The $T_{1,i}$ for different values of i are clearly iid since the codebook is independently drawn at each time t and the channel is memoryless.

- $T_{2,i}$: How long until the decoder is able to decode the codeword uniquely.

Lemma 7.1 of [4] applies to this term without list decoding. The $T_{2,i}$ are thus also iid across blocks i and

$$\mathcal{P}(T_{2,i} - \lceil \tilde{t}(\rho, R, n) \rceil ck_f > tck_f) \leq \exp(-tck_f E_0(C_f, \rho)) \quad (13)$$

for all $\rho \in [0, 1]$ and where $\tilde{t}(\rho, R, n) = \frac{R}{\tilde{C}(\rho)}n$ and $\tilde{C}(\rho) = \frac{E_0(C_f, \rho)}{\rho C_f \ln 2}$ after adjusting for the units of C_f bit packets used here for rate R .

- $T_{3,i}$: How long until the encoder realizes that the decoder has moved on.

The only way the encoder could miss this is if all the feedback packets were erased since the decoding succeeded. The only subtlety comes in measuring time. In keeping with tradition, in this section time is measured in forward channel uses and thus k_f and k_b are needed to translate.

$$\begin{aligned} \mathcal{P}(T_{3,i} > tk_f) &= (\beta_b)^{tk_b} \\ &= \exp(-tk_b(-\ln \beta_b)) \end{aligned}$$

and so the relevant exponent for T_3 is $-\frac{k_b}{k_f} \ln \beta_b$.

$T_{3,i}$ depends on the randomness in the feedback channel while $T_{1,i}$ and $T_{2,i}$ both depend on the forward channel and random codebooks, but at disjoint times. Thus, each of the terms are independent of each other since they depend on different independent random variables.

It is clear that the sum of three independent geometric random variables can be bounded by the slowest of them plus a constant. If two of them are equally slow, then the resulting polynomial growth term can be absorbed into a slightly smaller exponent. Thus for all $\epsilon > 0$, $\exists K$ depending on ϵ and the triple $(-\frac{k_b}{k_f} \ln \beta_b, E_0(C_f, \rho), E_0(C_f, 1))$ so that:

$$\mathcal{P}(T_i - \lceil \tilde{t}(\rho, R, n) \rceil ck_f - K > tck_f) \leq \exp(-tck_f(\min(E_0(C_f, \rho), -\frac{k_b}{k_f} \ln \beta_b) - \epsilon)) \quad (14)$$

since $E_0(C_f, \rho) \leq E_0(C_f, 1)$ because $\rho \in [0, 1]$ and the Gallager function E_0 is monotonically increasing in ρ . Notice also that the constant K here does not depend on n or c .

2) *Finishing the proof:* The conditions of Corollary 6.1 of [4] now apply with point messages arriving every nck_f channel uses, or every n units of time if time is measured in increments of ck_f channel uses. At this point, the proof proceeds in a manner identical to that of Theorem 3.4 in [4]. As long as ρ is chosen small enough so that $R < \frac{E_0(C_f, \rho)}{\rho C_f \ln 2}$, there exists n large enough so that $\tilde{t}(\rho, R, n) + \frac{K}{ck_f} < (1 - \delta)n$. The effective rate R'' from Corollary 6.1 of [4] is thus $\frac{1}{n - \tilde{t}(\rho, R, n) + \frac{K}{ck_f}} < \frac{1}{\delta n}$ point messages per ck_f forward channel uses. This can be made arbitrarily small by making n large and so by Theorem 3.3 in [4] the error exponent with end-to-end delay can be made arbitrarily close to $\min(E_0(C_f, \rho), -\frac{k_b}{k_f} \ln \beta_b)$. If the target error exponent α satisfies (1), then the minimum is known to be $E_0(C_f, \rho)$. This is maximized by increasing ρ so that $\frac{E_0(C_f, \rho)}{\rho C_f \ln 2}$ approaches R from above.

This demonstrates the asymptotic achievability of all reliability/rate points within the region obtained by varying $\rho \in [0, 1]$:

$$\begin{aligned} \alpha &< \min(-\frac{k_b}{k_f} \ln \beta_b, E_0(C_f, \rho)), \\ R &< \frac{E_0(C_f, \rho)}{\rho C_f \ln 2} \end{aligned} \quad (15)$$

Observe that ρC_f appears together in the expression for E_0 in (2) and so ρC_f plays the role of simple ρ for the binary erasure channel. Theorem 3.3 in [4] then gives the desired expression for the performance after converting from base 2 to base e .

The anytime property is inherited from Corollary 6.1 of [4]. \square

B. Theorem 2.2: with list decoding

When the forward and reverse channels have packet sizes of at least 2, it is possible to augment the protocol to use list-decoding to a list of size ℓ and some interaction to resolve the list ambiguity. The idea is to have the feedback encoder tell the forward encoder a subset of bit positions in the message block that it is confused about. For any pair of distinct messages, there exists a single bit position that would resolve the ambiguity between them. Since there are ℓ messages on the list, $\frac{\ell(\ell-1)}{2}$ such bit positions are clearly sufficient. Once the forward encoder knows which bit positions the decoder is uncertain about, it can communicate those particular bit values reliably using a repetition code.

The scheme is an extension of the scheme of Theorem 2.1 except with each message block requiring $m = 1 + \frac{\ell(\ell-1)}{2}(1 + \lceil \log_2 nck_f RC_f \rceil)$ rounds of communication instead of just 1 round. To support these multiple rounds, 1 bit is reserved on every forward packet and every feedback packet to carry the round number modulo 2.

These rounds have the following roles:

- 1 round: (Forward link leads, feedback link follows) A random codebook $X_i(j, t)$ as in the previous section is used by the forward encoder as before to communicate most of the information in the message block. The round stops when decoder has decoded to within ℓ possible choices of the codeword j . At that point, the feedback encoder will increment the round count in the feedback packets and initiate the next round.
- $\frac{\ell(\ell-1)}{2} \lceil \log_2 nck_f RC_f \rceil$ rounds: (Feedback link leads, forward link follows) The feedback encoder uses a repetition code to communicate $\frac{\ell(\ell-1)}{2}$ different bit positions within the block. Since there are $nck_f RC_f$ bits within a block, it takes $\lceil \log_2 nck_f RC_f \rceil$ bits to specify a specific bit position.

The feedback encoder uses the second bit in each 2-bit packet to carry the repetition code encoding these positions. As soon as the feedback channel is successful, the forward encoder will signal the round to advance by incrementing its counter. If this was the last such round, the forward encoder will initiate the next type of round. Otherwise, as soon as a forward channel is successful, the feedback encoder will also increment the round and move on to the next bit.

- $\frac{\ell(\ell-1)}{2}$ rounds: (Forward link leads, feedback link follows) The forward encoder uses a repetition code to communicate the specific values of the $\frac{\ell(\ell-1)}{2}$ requested bits. The rounds advance exactly as in the previous set of rounds.

Synchronization between the encoder and decoder is maintained because:

- They start out synchronized.
- The follower advances its counter as soon as it has decoded the round. As soon as the leader hears this, it too moves on to the next round. Because each packet comes with an unmistakable counter, it is interpreted correctly.

Because the channels are erasure channels, there is no possibility for confusion. Each follower advances to the next round only when it has learned what it needs from this one.

The proof that this achieves the desired error exponents is mostly parallel to that of the previous section. In the interests of brevity, only the differences are discussed here.

1) *The service time:* $T_i = T_{1,i} + \sum_{k=1}^{\frac{\ell(\ell-1)}{2} \lceil \log_2 nck_f RC_f \rceil} T_{2,i,k} + \sum_{k=1}^{\frac{\ell(\ell-1)}{2}} T_{3,i,k}$ since each round needs to complete for the entire block to complete.

- $T_{1,i}$: How long until the decoder is able to decode the codeword to within a list of size ℓ . This is almost the same as $T_{2,i}$ in the previous section. The only difference is that the effective forward packet size is $C_f - 1$ bits since 1 bit is reserved for the round number modulo 2.

Lemma 7.1 of [4] applies to this term with a list size of ℓ . The $T_{1,i}$ are thus iid across blocks i and

$$\mathcal{P}(T_{1,i} - \lceil \tilde{t}(\rho, R, n) \rceil ck_f > tck_f) \leq \exp(-tck_f E_0(C_f - 1, \rho)) \quad (16)$$

for all $\rho \in [0, \ell]$ and where $\tilde{t}(\rho, R, n) = \frac{R}{\tilde{C}(\rho)} n$ and $\tilde{C}(\rho) = \frac{E_0(C_f - 1, \rho)}{\rho C_f \ln 2}$ after adjusting for the notation used here including the units of C_f bit packets for rate R .

- $T_{2,i,k}$: How long it takes to complete one round of communicating a single bit from the feedback encoder to the forward encoder. This is the sum of two independent geometric random variables: one counting how long till a successful use of the feedback channel carrying the bit, and a second counting how long till a successful use of the forward channel carrying the confirmation that the bit was received.
- $T_{3,i,k}$: How long it takes to complete one round of communicating a single bit from the forward encoder. This is also the sum of the same two independent geometric random variables.

Use $T_f(k)$ to denote independent geometric (in increments of k_f) random variables counting how long it takes till a successful use of the forward channel. Similarly, use $T_b(k)$ for the backward channel in increments of k_b . Thus, $T = T_1 + \sum_{k=1}^{m-1} (T_f(k) + \frac{k_f}{k_b} T_b(k))$ has the distribution of the service time in terms of forward channel uses.

Clearly $E_0(C_f - 1, \rho) < -\ln \beta_f$ for all $\rho > 0$. There are two possibilities depending on whether T_1 provides the dominant error exponent: $E_0(C_f - 1, \rho^*) < -\frac{k_b}{k_f} \ln \beta_b$ for the ρ^* that solves $\tilde{C}(\rho^*) = R$. If the feedback channel provides the dominant exponent, set $\rho < \rho^*$ so that $E_0(C_f - 1, \rho) = -\frac{k_b}{k_f} \ln \beta_b$. Otherwise, leave $\rho < \rho^*$ free for now. Define $\gamma = E_0(C_f - 1, \rho)$ as the dominant exponent.

Let $T'(k)$ be iid geometric random variables (in increments of ck_f) that are governed by the exponent γ so $\mathcal{P}(T' > tck_f) = \exp(-tck_f \gamma)$. Consider $\sum_{k=1}^{2m-1} T'(k)$. This has a negative binomial or Pascal distribution.

Lemma 3.1: Let $T'(k)$ be iid geometric random variables that are governed by the exponent γ so $\mathcal{P}(T' > t) = \exp(-t\gamma)$. Then, for every $\epsilon' > 0$, there exists an $\epsilon > 0$ that depends only on γ and ϵ' so that $\forall t > 0$:

$$\mathcal{P}\left(\sum_{k=1}^{2m-1} T'(k) > t + \tilde{t}\right) < \exp(-\gamma(1 - \epsilon')(t + \tilde{t})) \quad (17)$$

where $\tilde{t} = \frac{2m-1}{\epsilon}$.

Proof: See Appendix A.

This means that the service time T_i has a complementary CDF that is bounded by:

$$\mathcal{P}(T_i - (\lceil \tilde{t}(\rho, R, n) \rceil + \check{t})ck_f > tck_f) < \exp(-tck_f(1 - \epsilon')E_0(C_f - 1, \rho)). \quad (18)$$

2) *Finishing the proof:* The proof can continue almost the same way as in the previous section. All that needs to be checked is that for a given target error exponent $(1 - \epsilon')E_0(C_f - 1, \rho)$, the overhead $\lceil \tilde{t}(\rho, R, n) \rceil + \check{t}$ can be made smaller than n so that the point message rate R'' in Corollary 6.1 of [4] can be made to go to zero.

Assuming that $\ell \geq 2$ (otherwise what is the point of using list-decoding!):

$$\begin{aligned} \lceil \tilde{t}(\rho, R, n) \rceil + \check{t} &\leq 1 + \frac{R}{\tilde{C}(\rho)}n + \frac{2 + 2^{\frac{\ell(\ell-1)}{2}}(1 + \lceil \log_2 nck_fRC_f \rceil)}{\epsilon} \\ &\leq \left[\frac{R}{\tilde{C}(\rho)} + \frac{\ell(\ell-1)(3 + \log_2 ck_fRC_f)}{\epsilon} \left(\frac{\log_2 n}{n} \right) \right] n. \end{aligned}$$

Clearly whenever $\tilde{C}(\rho) > R$, there exists an n big enough so that the entire term in brackets $[\dots] < 1 - \delta$ for some small $\delta > 0$. From this point on, the proof proceeds exactly as before. Recall that ϵ' is arbitrary so this gets us asymptotically to the fixed-delay reliability region parametrized as:

$$\begin{aligned} \alpha &< \min(E_0(C_f - 1, \rho), -\frac{k_b}{k_f} \ln \beta_b), \\ R &< \frac{E_0(C_f - 1, \rho)}{\rho C_f \ln 2} \end{aligned} \quad (19)$$

where ρ ranges from 0 to ℓ . But ℓ can be chosen as high as needed. Finally, the rate in (19) can be rewritten as $R < \frac{C_f - 1}{C_f} \left(\frac{E_0(C_f - 1, \rho)}{\rho(C_f - 1) \ln 2} \right)$. Notice that $(C_f - 1)\rho$ appear together in the expression (2) for $E_0(C_f - 1, \rho)$ in the place of the simple ρ for the binary erasure channel. This lets us use Theorem 3.3 in [4] to get the desired expression, once again doing the straightforward conversions from base 2 to base e . \square

C. Extensions

While packet-erasure channels were considered for concreteness of exposition, it is clear that Theorem 2.1 extends to any channel on the forward link for which the zero-undetected-error³ capacity equals the regular capacity (See Problem 5.32 in [31]). If the probability of undetected error is zero, then decoding proceeds by eliminating codewords as being impossible. That is all that is needed in this proof. In particular, the result extends immediately to packet-valued channels that can erase individual bits within packets according to some joint distribution rather than having to erase only the entire packet or nothing at all.

Similarly on the feedback path, the proof of Theorem 2.1 only requires the ability to carry a single bit message unambiguously in a random amount of time, where that time has a distribution that is bounded by a geometric. For a channel whose zero-undetected-error capacity equals the regular capacity, a random code can be used but with only two codewords. This gives an exponent of at least $E_0^b(1)$ on the feedback path. Therefore, the arguments of this section have essentially already proved the following theorem showing optimality at high rates for more general channels:

Theorem 3.1: Consider the (k_f, k_b) problem of Figures 1 and 2 with $k_f, k_b \geq 1$, and forward DMC P_f and backward DMC P_b , both with their zero-undetected-error capacities (without feedback⁴) equal to their regular Shannon capacities. Suppose that $k_r > 0$ is the round-trip delay (measured in cycles).

In the limit of large end-to-end delays, it is possible to asymptotically achieve all (R, α) pairs

$$\begin{aligned} \alpha &< \min \left(-\frac{k_b}{k_f} E_0^b(1), E_0^f(\rho) \right), \\ R &< \frac{E_0^f(\rho)}{\rho} \end{aligned} \quad (20)$$

³Zero-undetected-error means that the probability of error is zero if the decoder is also allowed to refuse to decode. For capacity to be meaningful, the probability of such refusals must approach zero as the delay or block-length gets large.

⁴Perfect feedback increases the zero-undetected-error capacity all the way to the Shannon capacity for DMCs. A system can just use a one bit message at the end to tell the decoder whether or not to accept its tentative decoding [32].

in an anytime fashion where E_0^f and E_0^b are the Gallager functions for the forward and reverse channels respectively and $\rho \in [0, 1]$. The rate R above is measured in nats per forward channel use as is the reliability α .

Proof: To deal with the round-trip delay, just extend the cycle length by considering $k'_f = \frac{1}{\epsilon} k_f k_r$, $k'_b = \frac{1}{\epsilon} k_b k_r$. Consider the last $k_f k_r$ channel uses of each extended cycle to be wasted. The effective number of forward channel uses is thus reduced by a factor $(1 - \epsilon)$. Since this factor reduction can be made as small we like, asymptotically, the problem reduces to the case with no round-trip-delay.

To patch the proof of Theorem 2.1 to account for the general channels on the forward and feedback links:

- Use the $E_0^f(\rho^*)$ optimizing input distribution $\vec{q}(\rho^*)$ for the random forward channel codebooks. $\rho^* \in [0, 1]$ is chosen so that the target $\alpha < E_0^f(\rho^*)$ and $R < \frac{E_0^f(\rho^*)}{\rho^*}$.
- Use the $E_0^b(1)$ optimizing input distribution for the random two-codeword codebooks on the feedback channel.
- The analysis of $T_{1,i}$ is unchanged since $E_0^f(1, \vec{q}(\rho^*)) > E_0^f(\rho^*)$ by the properties of the Gallager function [30].
- The analysis of $T_{2,i}$ is entirely unchanged and gives $E_0^f(\rho^*)$ as the relevant exponent.
- The analysis of $T_{3,i}$ is now exactly parallel to $T_{1,i}$ since this also succeeds the instant the two random codewords on the feedback link can be distinguished by the received symbol. This is governed by the exponent $E_0^b(1)$ in terms of feedback channel uses and thus $\frac{k_b}{k_f} E_0^b(1)$ in terms of forward channel uses.

Everything else proceeds identically, except with rate units in nats per forward channel use rather than in normalized units of C_f bits. \square

It is unclear how to extend Theorem 2.2 to these general erasure-style channels. To break the $E_0(1)$ barrier, the construction in Theorem 2.2 relies on having a single header bit that always shows up. This approach does extend to the packet-truncation channels of [33] by making the header bit come first, but it does not extend to erasure-style channels in which individual bits within a packet can be erased in some arbitrary fashion.

The restriction to channels whose zero-undetected-error capacity without feedback *equals* their Shannon capacity is quite strict, and is required for the above schemes to work. This allows us to use imperfect feedback since the decoder can be counted on to know when to stop on its own. However, the approach of Section III-B can be used to extend Theorem 3.4 of [4] when the feedback is perfect. Recall that Theorem 3.4 of [4] requires a zero-error capacity that is strictly greater than zero. The multi-round approach with repetition codes allows us to drop this condition and merely require that the zero-undetected-error capacity without feedback be strictly greater than zero (*ie* the channel matrix P has at least one zero entry in a row and column that is not identically zero).

Theorem 3.2: For any DMC whose transition matrix P contains a nontrivial zero, it is possible to use noiseless perfect feedback and randomized encoders to asymptotically approach all delay exponents within the region

$$\begin{aligned} \alpha &< \min(E_0(\rho), E^*), \\ R &< \frac{E_0(\rho)}{\rho} \end{aligned} \tag{21}$$

where $E_0(\rho)$ is the Gallager function, ρ ranges from 0 to ∞ , and E^* is the error exponent governing the zero-undetected-error transmission of a single bit message.

Furthermore, the delay exponents α can be achieved in a delay-universal or “anytime” sense even if the feedback is delayed by an amount ϕ that is small relative to the asymptotically large target end-to-end delay.

Proof: The proof and overall approach is almost identical to what has done previously. Only the relevant differences will be covered here.

First, it is well known that the presence of a nontrivial zero makes the zero-undetected-error capacity strictly positive. To review, let x, x' be input letters so that there exists a y such $\mathcal{P}(Y = y|X = x) = 0$ while $\mathcal{P}(Y = y|X = x') > 0$. Then the following two mini-codewords can be used: (x, x') and (x', x) . The probability of unambiguous decoding is at least $\mathcal{P}(Y = y|X = x')$ and so by using this as a repetition code, the error exponent $E^* \geq \frac{1}{2} \ln(1 - \mathcal{P}(Y = y|X = x')) > 0$ per forward channel use. Of course, it can be much higher depending on the specific channel.

The (n, c, l) scheme of [4] is modified as follows:

- Each chunk of c channel uses is now itself implemented as variable length. It consists of a fixed c_f channel uses that are used exactly as before to carry a part of a random codeword. To this is appended a variable-length code that uses perfect feedback to communicate exactly 1 bit without error. This bit consists of the

“punctuation” telling the decoder whether or not there is a comma after this chunk (*ie* whether the decoder’s tentative list-decoding contains the true codeword or not).

- If the list size $\ell = 2^l \neq 1$, then the l bits of list-disambiguation information are conveyed by using l successive single-bit variable-length codes before the next block begins.

Perfect noiseless feedback is assumed as in [4] so that the forward encoder knows when to stop each round and move on. No headers are required. The main difference from Theorem 2.2 is that the number of rounds required to communicate a message block is not fixed. Instead, each message takes a variable number of rounds.

The idea is to choose $c - c_f$ large and then to pick an effective c_f that is so big that it is almost proportional to c . Let T_i be the total service time for the i -th message block as measured in forward channel uses. It is clear that this is the sum of

- $T_{1,i}c_f$: The number of forward channel uses required for the random codeword before the message can be correctly list-decoded to within ℓ possibilities. This is exactly as before and is governed by Lemma 7.1 of [4].
- $\sum_{k=1}^{T_{1,i}} T_{2,i,k}$: The number of forward channel uses required to communicate the $T_{1,i}$ distinct punctuation symbols in the block. Each individually is governed by the exponent E^* . This is the qualitatively new term.
- $\sum_{k=1}^l T_{3,i,k}$: The number of forward channel uses required to communicate all of the l distinct 1-bit disambiguation messages. These are also governed by the exponent E^* .

It is easiest to upper-bound the complementary CDF for $T_{1,i}c_f + \sum_{k=1}^{T_{1,i}} T_{2,i,k} + \sum_{k=1}^l T_{3,i,k}$ together. Define $\epsilon = \frac{1}{c-c_f}$, $\tilde{t}(\rho, R, n) = \lceil \frac{R}{C(\rho)} n \rceil$.

$$\begin{aligned}
& \mathcal{P}(T_{1,i}c_f + \sum_{k=1}^{T_{1,i}} T_{2,i,k} + \sum_{k=1}^l T_{3,i,k} - \tilde{t}(\rho, R, n)c > tc) \\
\leq_a & \mathcal{P}(T_{1,i}c_f - \tilde{t}(\rho, R, n)c_f \geq tc_f) \\
& + \sum_{s=1}^t (\mathcal{P}(T_{1,i}c_f - \tilde{t}(\rho, R, n)c_f = (t-s)c_f)) \left(\mathcal{P}\left(\sum_{k=1}^{\tilde{t}(\rho, R, n)+t-s+l} T_{2,i,k} > \tilde{t}(\rho, R, n)(c-c_f) + (t-s)(c-c_f) + sc \right) \right) \\
\leq_b & \exp(-tc_f E_0(\rho)) + \sum_{s=1}^t \exp(-(t-s)c_f E_0(\rho)) \left(\mathcal{P}\left(\sum_{k=1}^{\tilde{t}(\rho, R, n)+t-s+l} T_{2,i,k} > (\tilde{t}(\rho, R, n) + t-s)(c-c_f) + sc \right) \right) \\
= & \exp(-tc_f E_0(\rho)) + \sum_{s=1}^t \exp(-(t-s)c_f E_0(\rho)) \left(\mathcal{P}\left(\sum_{k=1}^{\tilde{t}(\rho, R, n)+t-s+l} T_{2,i,k} > \frac{\tilde{t}(\rho, R, n) + t-s+l}{\epsilon} + sc - \frac{l}{\epsilon} \right) \right) \\
<_c & \exp(-tc_f E_0(\rho)) + \sum_{s=1}^t \exp(-(t-s)c_f E_0(\rho)) \left(\exp(-(1-\epsilon') \left(\frac{\tilde{t}(\rho, R, n) + t-s+l}{\epsilon} + sc - \frac{l}{\epsilon} \right) E^*) \right) \\
=_{d} & \exp(-tc_f E_0(\rho)) + \left(\exp(-(1-\epsilon') \frac{\tilde{t}(\rho, R, n) + t}{\epsilon} E^*) \right) \sum_{s=1}^t \exp(-(t-s)c_f E_0(\rho)) \exp(-(1-\epsilon')sc_f E^*) \\
<_e & 2 \exp(-tc_f \min(E_0(\rho), (1-\epsilon')E^*)) \\
=_{f} & 2 \exp\left(-tc \frac{c_f}{c} \min(E_0(\rho), (1-\epsilon')E^*)\right).
\end{aligned}$$

where (a) is a union bound over different ways that the budget of $\tilde{t}(\rho, R, n)c + tc$ channel uses could be exceeded together with the independence of the different component service times. Notice that all the terms governed by E^* are folded in together. (b) comes from simple algebra together with applying Lemma 7.1 of [4] and is valid as long as $\rho \leq \ell$. (c) is the result of substituting in the definition of ϵ and then applying Lemma 3.1. (d) brings out the sc_f term in the second term and then (e) reflects that the sum of exponentials is dominated by the largest term. (f) is a simple renormalization to c units so that the result is plug-in compatible with Lemma 7.1 of [4].

Choosing c_f large enough tells us that for all $\epsilon'' > 0$, there exist c, c_f large enough so that:

$$\mathcal{P}(T_j - \tilde{t}(\rho, R, n)c > tc) \leq 2[\exp(-c(1-\epsilon'') \min(E_0(\rho), E^*))]^t \quad (22)$$

as long as $\rho \in [0, \ell]$. From this point onward, the proof is identical to the original in [4]. \square

IV. SPLITTING A SHARED RESOURCE BETWEEN THE FORWARD AND FEEDBACK CHANNEL: THEOREM 2.3

The goal of this section is to prove Theorem 2.3 by considering what the best choices for k_f and k_b are if both the rate and delay are considered relative to the sum $k_f + k_b$ rather than just forward channel uses alone.

A. Evaluating the previous schemes

Assume k_f and k_b are fixed and let $\eta_f = \frac{k_f}{k_f + k_b}$ and $\eta_b = \frac{k_b}{k_f + k_b}$. The η_f acts as the conversion factor mapping both the error exponents and the rates from per-forward-channel-use units to per-total-channel-uses units. Similarly, let $\xi_f = \frac{k_f C_f}{k_f C_f + k_b C_b}$ and $\xi_b = \frac{k_b C_b}{k_f C_f + k_b C_b}$. The ξ_f is the conversion factor that maps error exponents and rates to per-weighted-total-channel-uses units.

Thus for Theorem 2.1, (15) becomes

$$\alpha' < \min(-\eta_b \ln \beta_b, \eta_f E_0(C_f, \rho)), \quad (23)$$

$$\bar{\alpha} < \min\left(-\frac{k_b C_b}{k_b C_b + k_f C_f} \ln \beta_b, \xi_f E_0(C_f, \rho)\right), \quad (24)$$

$$R' < \frac{\eta_f E_0(C_f, \rho)}{\rho C_f \ln 2}, \quad (25)$$

$$\bar{R} < \frac{\xi_f E_0(C_f, \rho)}{\rho C_f \ln 2}. \quad (26)$$

For Theorem 2.2, the range of ρ expands to $\rho \in [0, \infty)$ but the rate terms change to

$$R' < \frac{\eta_f E_0(C_f - 1, \rho)}{\rho C_f \ln 2}, \quad (27)$$

$$\bar{R} < \frac{\xi_f E_0(C_f - 1, \rho)}{\rho C_f \ln 2}. \quad (28)$$

B. Optimizing by adjusting k_f and k_b

If $-\eta_b \ln \beta_b > \eta_f E_0(C_f, \rho)$, then it is the forward link that is the bottleneck. If the inequality is in the opposite direction, then the feedback link is what is limiting reliability. This suggests that setting the two exponents equal to each other gives a good exponent α' . Since $\eta_b = 1 - \eta_f$, this means $\eta_f^* = \frac{-\ln \beta_b}{E_0(C_f, \rho) - \ln \beta_b}$. Plugging this in reveals that all α' are achievable that satisfy:

$$\alpha' < \frac{(-\ln \beta_b) E_0(C_f, \rho)}{E_0(C_f, \rho) - \ln \beta_b} \quad (29)$$

$$= \left(\frac{E_0(C_f, \rho) - \ln \beta_b}{(-\ln \beta_b) E_0(C_f, \rho)}\right)^{-1} \quad (30)$$

$$= E_0'(C_f, \rho). \quad (31)$$

Plugging in for R' reveals that all the R' that satisfy

$$R' < \frac{\eta_f E_0(C_f, \rho)}{\rho C_f \ln 2} \quad (32)$$

$$= \frac{E_0'(C_f, \rho)}{\rho C_f \ln 2} \quad (33)$$

are also achievable. This establishes (6) and identical arguments give (8).

To see what happens when C_f gets large while C_b stays constant, just notice that in such a case $\xi_f = \frac{k_f C_f}{k_f C_f + k_b C_b}$ gets close to 1 no matter how big k_b is. This establishes the α', \bar{R} tradeoff in (9).

Alternatively, k_b can be chosen to be large enough so that $-\frac{k_b C_b}{k_b C_b + k_f C_f} \ln \beta_b > -\xi \ln \beta_f > \xi_f E_0(C_f, \rho)$. Then taking $C_f \rightarrow \infty$ immediately gives the desired $\bar{\alpha}, \bar{R}$ tradeoff. \square

Theorem 2.3 can clearly be extended to the general setting of Theorem 3.1. The relevant $E'_0(\rho)$ is immediately seen to be

$$E'_0(\rho) = \left(((E_0^b(1))^{-1} + (E_0^f(\rho))^{-1}) \right)^{-1}. \quad (34)$$

The parallel to Theorem 3.5 of [4] is obvious and makes sense since both involve splitting a shared resource to two purposes that must be balanced. Here, it is channel uses across the feedback and forward channels. In Theorem 3.5 of [4], it is allocating forward channel uses to carrying messages and flow control information.

V. CONCLUSIONS

It has been shown that in the limit of large end-to-end delays, perfect feedback performance is attainable by using appropriate random codes at high rate for erasure channels even if the feedback channel is an unreliable erasure channel. Somewhat surprisingly, this does not require any explicit header bits on the packets if the rate is high enough and thus works even for a system with a BEC in the forward link and a BEC in the feedback link. The reliability gains from using feedback are so large that they persist even when each feedback channel use comes at the cost of not being able to use the forward channel (half-duplex). This was shown by considering both rate and delay in terms of total channel uses rather than just the forward channel uses.

The arguments here readily generalize to all channels for which the zero-undetected-error capacity equals the regular capacity, but do not extend to channels like the BSC. Even when the zero-undetected-error capacity is strictly larger than zero, the techniques here just give an improved result for the case of perfect feedback. Showing that the gains from feedback are robust to unreliable feedback in such cases remains an open problem. In addition, the results here are on the achievability side. The best upper-bounds to reliability in the fixed end-to-end delay context are still those from [4] and it remains an open problem to tighten the bounds when the feedback is unreliable or in the half-duplex situation.

APPENDIX A

PASCAL DISTRIBUTION BOUND: LEMMA 3.1

Consider $\sum_{k=1}^{2m-1} T'(k)$ where the $T'(k)$ are iid geometric random variables with exponent γ . This has a negative binomial or Pascal distribution. The probability distribution of this sum is easily bounded by interpreting the Pascal distribution as the $(2m-1) - th$ arrival time of a virtual $\{Z_k\}$ Bernoulli process with probability of failure $\exp(-\gamma c k_f)$. Pick an $\epsilon > 0$.

It is clear that

$$\begin{aligned} \mathcal{P}\left(\sum_{k=1}^{2m-1} T'(k) - \check{t} > t\right) &= \mathcal{P}\left(\sum_{k=1}^{t+\check{t}} Z_k < 2m-1\right) \\ &= \mathcal{P}\left(\frac{\sum_{k=1}^{t+\check{t}} Z_k}{t+\check{t}} < \frac{2m-1}{t+\check{t}}\right) \\ &\leq \mathcal{P}\left(\frac{\sum_{k=1}^{t+\check{t}} Z_k}{t+\check{t}} < \frac{2m-1}{\check{t}}\right) \\ &= \mathcal{P}\left(\frac{\sum_{k=1}^{t+\check{t}} Z_k}{t+\check{t}} < \epsilon\right) \\ &\leq (t+\check{t}) \exp(-(t+\check{t})D(1-\epsilon|\exp(-\gamma))). \end{aligned}$$

But when ϵ is small,

$$\begin{aligned} D(1-\epsilon|\exp(-\gamma)) &= (1-\epsilon)(\ln(1-\epsilon) + \gamma) + \epsilon \ln\left(\frac{1}{1-\exp(-\gamma)}\right) + \ln \epsilon \\ &\geq (1-\epsilon)(\gamma - 2\epsilon) + \epsilon \ln \epsilon \\ &= (1-\epsilon)\gamma - \epsilon \ln \frac{1}{\epsilon} + 2(1-\epsilon). \end{aligned}$$

So:

$$\begin{aligned}
\mathcal{P}\left(\sum_{k=1}^{2m-1} T'(k) - \tilde{t} > t\right) &\leq (t + \tilde{t}) \exp(-(t + \tilde{t})D(1 - \epsilon \|\exp(-\gamma))) \\
&\leq \exp(-(t + \tilde{t})[(1 - \epsilon)\gamma - \epsilon(\ln \frac{1}{\epsilon} + 2(1 - \epsilon)) - \frac{\ln(t + \tilde{t})}{t + \tilde{t}}]) \\
&< \exp(-(t + \tilde{t})[(1 - \epsilon)\gamma - \epsilon(\ln \frac{1}{\epsilon} + 2) - \frac{\ln \tilde{t}}{\tilde{t}}]) \\
&= \exp(-(t + \tilde{t})[(1 - \epsilon)\gamma - \epsilon(\ln \frac{1}{\epsilon} + 2 + \frac{\ln(2m - 2) + \ln \frac{1}{\epsilon}}{2m - 2})]) \\
&< \exp(-(t + \tilde{t})[(1 - \epsilon)\gamma - \epsilon(2 \ln \frac{1}{\epsilon} + 3)])
\end{aligned}$$

As $\epsilon \rightarrow 0$, the term $\epsilon(2 \ln \frac{1}{\epsilon} + 3)$ also vanishes. So for any $\epsilon' > 0$, we can choose ϵ small enough so that $(1 - \epsilon)\gamma - \epsilon(2 \ln \frac{1}{\epsilon} + 3) > (1 - \epsilon')\gamma$. This gives:

$$\begin{aligned}
\mathcal{P}\left(\sum_{k=1}^{2m-1} T'(k) - \tilde{t} > t\right) &< \exp(-(t + \tilde{t})[(1 - \epsilon)\gamma - \epsilon(2 \ln \frac{1}{\epsilon} + 3)]) \\
&< \exp(-(t + \tilde{t})(1 - \epsilon')\gamma)
\end{aligned}$$

This completes the proof of Lemma 3.1. □

ACKNOWLEDGMENTS

We thank the NSF ITR program (ANI-0326503) for funding this research.

REFERENCES

- [1] C. E. Shannon, "The zero error capacity of a noisy channel," *IEEE Trans. Inf. Theory*, vol. 2, no. 3, pp. 8–19, Sep. 1956.
- [2] R. L. Dobrushin, "An asymptotic bound for the probability error of information transmission through a channel without memory using the feedback," *Problemy Kibernetiki*, vol. 8, pp. 161–168, 1962.
- [3] E. R. Berlekamp, "Block coding with noiseless feedback," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, 1964.
- [4] A. Sahai, "Why block-length and delay behave differently if feedback is present," *IEEE Trans. Inf. Theory*, Submitted. [Online]. Available: <http://www.eecs.berkeley.edu/~sahai/Papers/FocusingBound.pdf>
- [5] M. S. Pinsker, "Bounds on the probability and of the number of correctable errors for nonblock codes," *Problemy Peredachi Informatsii*, vol. 3, no. 4, pp. 44–55, Oct./Dec. 1967.
- [6] R. W. Lucky, "A survey of the communication theory literature: 1968–1973," *IEEE Trans. Inf. Theory*, vol. 19, no. 5, pp. 725–739, Nov. 1973.
- [7] L. J. Schulman, "Coding for interactive communication," *IEEE Trans. Inf. Theory*, vol. 42, no. 6, pp. 1745–1756, Nov. 1996.
- [8] S. Rajagopalan, "A coding theorem for distributed computation," Ph.D. dissertation, University of California, Berkeley, 1994.
- [9] J. P. M. Schalkwijk and T. Kailath, "A coding scheme for additive noise channels with feedback – I: No bandwidth constraint," *IEEE Trans. Inf. Theory*, vol. 12, no. 2, pp. 172–182, Apr. 1966.
- [10] J. P. M. Schalkwijk, "A coding scheme for additive noise channels with feedback – II: Band-limited signals," *IEEE Trans. Inf. Theory*, vol. 12, no. 2, pp. 183–189, Apr. 1966.
- [11] R. L. Kashyap, "Feedback coding schemes for an additive noise channel with a noisy feedback link," *IEEE Trans. Inf. Theory*, vol. 14, pp. 471–480, May 1968.
- [12] A. J. Kramer, "Improving communication reliability by use of an intermittent feedback channel," *IEEE Trans. Inf. Theory*, vol. 15, no. 1, pp. 52–60, Jan. 1969.
- [13] S. S. Lavenburg, "Feedback communication using orthogonal signals," *IEEE Trans. Inf. Theory*, vol. 15, no. 4, pp. 478–483, Jul. 1969.
- [14] —, "Repetitive signaling using a noisy feedback channel," *IEEE Trans. Inf. Theory*, vol. 17, no. 3, pp. 269–278, May 1971.
- [15] T. Weissman, A. Lapidoth, and Y. H. Kim, "Robustness and sensitivity of the schalkwijk-kailath scheme," Tech. Rep., Jul. 2006.
- [16] N. Martins and T. Weissman, "A coding scheme for additive white noise channels with feedback corrupted by quantization or bounded noise," Tech. Rep., Jul. 2006.
- [17] A. Sahai and S. K. Mitter, "The necessity and sufficiency of anytime capacity for stabilization of a linear system over a noisy communication link. part I: scalar systems," *IEEE Trans. Inf. Theory*, vol. 52, no. 8, pp. 3369–3395, Aug. 2006.
- [18] M. V. Burnashev, "Data transmission over a discrete channel with feedback, random transmission time," *Problemy Peredachi Informatsii*, vol. 12, no. 4, pp. 10–30, Oct./Dec. 1976.
- [19] H. Yamamoto and K. Itoh, "Asymptotic performance of a modified Schalkwijk-Barron scheme for channels with noiseless feedback," *IEEE Trans. Inf. Theory*, vol. 25, no. 6, pp. 729–733, Nov. 1979.

- [20] A. Sahai and T. Simsek, "On the variable-delay reliability function of discrete memoryless channels with access to noisy feedback," in *Proceedings of the IEEE Workshop on Information Theory*, San Antonio, TX, Nov. 2004.
- [21] S. Draper, K. Ramchandran, B. Rimoldi, A. Sahai, and D. Tse, "Attaining maximal reliability with minimal feedback via joint channel-code and hash-function design," in *Proceedings of the Allerton Conference on Communication, Control, and Computing*, Monticello, IL, Sep. 2005.
- [22] S. Draper and A. Sahai, "Noisy feedback improves communication reliability," in *Proc. IEEE International Symposium on Information Theory*, Jul. 2006. [Online]. Available: <http://www.eecs.berkeley.edu/~sahai/Papers/DraperSahaiISIT06.pdf>
- [23] —, "Variable-length coding with noisy feedback," *European Transactions on Telecommunications*, Submitted.
- [24] A. Sahai and S. Draper, "Beating the Burnashev bound using noisy feedback," in *Proceedings of the Allerton Conference on Communication, Control, and Computing*, Monticello, IL, Sep. 2006.
- [25] B. D. Kudryashov, "Message transmission over a discrete channel with noiseless feedback," *Problemy Peredachi Informatsii*, vol. 15, no. 1, pp. 1–9, Jan. 1979.
- [26] J. Massey, "Feedback, erasures, zero-errors, and more," Tech. Rep., Jan. 2006. [Online]. Available: <http://webcast.rice.edu/index.php?action=details&event=610>
- [27] L. Subramanian, I. Stoica, H. Balakrishnan, and R. Katz, "OverQoS: An overlay based architecture for enhancing internet QoS," in *Proceedings of the 1st Symposium on Networked Systems Design and Implementation (NSDI)*, San Francisco, CA, Mar. 2004.
- [28] F. Jelinek, "Upper bounds on sequential decoding performance parameters," *IEEE Trans. Inf. Theory*, vol. 20, no. 2, pp. 227–239, Mar. 1974.
- [29] G. D. Forney, "Convolutional codes II. maximum-likelihood decoding," *Information and Control*, vol. 25, no. 3, pp. 222–266, Jul. 1974.
- [30] R. G. Gallager, *Information Theory and Reliable Communication*. New York, NY: John Wiley, 1971.
- [31] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems*. New York: Academic Press, 1981.
- [32] G. D. Forney, "Exponential error bounds for erasure, list, and decision feedback schemes," *IEEE Trans. Inf. Theory*, vol. 14, pp. 206–220, 1968.
- [33] N. C. Martins, N. Elia, and M. A. Dahleh, "Stabilization of uncertain systems in the presence of a stochastic digital link," *IEEE Trans. Autom. Control*, submitted for publication.