Multiplexed FEC for Multiple Streams with Different Playout Deadlines

Ahmed Badr, *Member, IEEE*, Devin Lui, Ashish Khisti, *Member, IEEE*, Wai-tian Tan, *Member, IEEE*, Xiaoqing Zhu, *Member, IEEE* and John Apostolopoulos, *Fellow IEEE*

***Abstract*— We consider a streaming setup where two source streams with different decoding deadlines, must be transmitted over a single channel subjected to burst erasures. The encoder multiplexes the two source streams into a single stream of channel-packets. The decoder must recover the source-packets sequentially by their corresponding deadlines. The packet belonging to one of the streams — referred to as the urgent stream — will have a smaller delay than the other stream.**

We show that in general that is a tradeoff between the rates of the two source streams and characterize the capacity region for a certain range of system parameters. On the achievability side, we provide new code constructions which combine the source-packets of the two streams despite their different deadlines. On the converse side, we develop rigorous information theoretic outer bounds on the capacity region.

Interestingly we find that the capacity region exhibits a "corner point" where we can transmit the urgent stream at a positive rate, yet attain a sum-rate equal to the capacity of the non-urgent stream. We show that a baseline scheme which applies a single stream code separately to each of the streams is suboptimal in general.

I. Introduction

A growing number of multimedia applications including video conferencing, cloud computing, and mobile gaming are inherently *streaming* in nature. These applications must operate in real-time and under strict delay constraints. Such systems are susceptible to sporadic burst packet losses and long packet delays in wireless networks. While forward error correction (FEC) codes provide a natural mechanism for combating such losses, streaming applications discussed above impose some unique constraints not satisfied by traditional FEC. In these applications both the encoding and decoding operations must operate sequentially on the source and channel streams respectively. Unlike traditional FEC that operate over long block lengths, typically consisting of a few thousand packets or more, FEC in streaming applications must operate over much shorter blocks due to the delay constraints. Such codes are particularly sensitive when packet losses happen in bursts [1], [2]. It turns out that local distance properties, as opposed to global metrics such as minimum distance, are much more significant in determining the performance of error control codes in streaming applications.

A new class of error correction codes for streaming applications is introduced in [1]. The encoder observes a semi-infinite source stream — one packet is revealed in each time slot — and maps it to a coded output stream of rate R. The channel is modeled as a burst-erasure channel. Starting at an arbitrary time, it introduces an erasure-burst of maximum length B. The decoder is required to reconstruct each source-packet with a maximum delay T. A fundamental relationship between R, B and T is established and a novel construction is proposed that achieves it. These codes are systematic, linear, time-invariant, convolutional codes where the parity check symbols involve a careful combination of source symbols. In particular, random linear combinations, popularly used in e.g., network coding, do not attain the optimal performance. In reference [2] these constructions are extended to channels with both burst and isolated erasures. A fundamental tradeoff between the column-distance and column-span for a convolutional code is introduced and its operational relevance in terms of error correction in the streaming setup is established. The constructions proposed in [2] provide considerable gains over baseline schemes in simulations over statistical models such as the Gilbert-Elliott channel. We refer the reader to [2]–[13] and the references therein for various extensions of these works. We note that in the broader literature a number of approaches for streaming have been considered, see e.g., [14]–[21] and references therein. However these works are not directly related to the present formulation.

In the present work we study a streaming setup where two source streams with different delays must be simultaneously encoded into a single channel-packet stream. Such a scenario arises naturally in video streaming. For example in corporate video conferencing, one stream could comprise of the audio/video content and will require a stringent delay constraint, while the other stream comprising of presentation slides could have a more relaxed delay constraint. As another example, consider a relay node that must multiplex two incoming source streams and transmit over a common communication link. The two source streams could have different delay requirements either due to application constraints or networking effects such as link delays/packet-losses over previous hops. As a final motivation a recent proposal on Quick UDP internet connection (QUIC) aims to multiplex multiple streams for reliable transport over UDP connections [22], and joint FEC across multiple streams is a natural solution for such applications.

In this paper we initiate the study of such a setup. We focus

Ahmed Badr, Devin Lui and Ashish Khisti are with the School of Electrical and Computer Engineering, University of Toronto, Toronto, ON, M5S 3G4 Canada (e-mail: {abadr, dlui, akhisti}@ece.utoronto.ca). Wai-tian Tan, Xiaoqing Zhu and John Apostolopoulos are with Chief Technology and Architecture Office , Cisco Systems, San Jose, CA, 95134 (e-mail: {dtan2, xiaoqzhu, johnapos}@cisco.com).

This work was supported by a discovery grant from National Science and Engineering Research Council of Canada, Cisco Systems, and an Ontario Early Researcher Award. Preliminary results of this work were presented at the 2016 Workshop on Network Coding and Applications, Washington, DC.

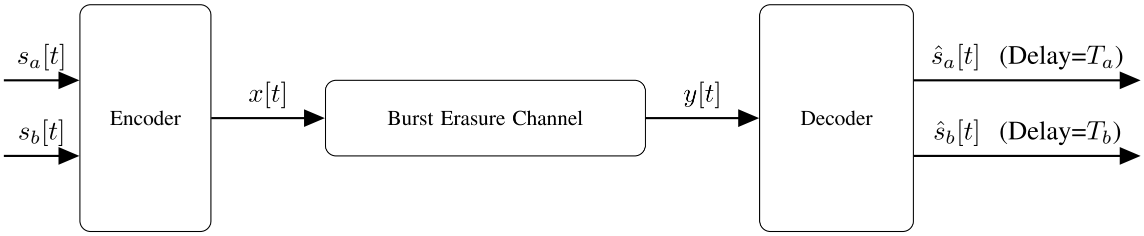


Fig. 1: Streaming Setup with Multiplexed Flows. There are two source streams a and b, which are multiplexed into a single channel-packet stream. The source streams are recovered at the decoder with different delays.

on the burst-erasure channel model. We show that by jointly coding both the streams, despite their different deadlines, we can achieve higher rates than the baseline scheme that applies streaming codes separately on the two streams. We show that the capacity region exhibits a "corner point" where we can transmit the more urgent stream at a certain positive rate, while keeping the sum rate to equal the capacity of the stream with larger delay. We also establish information theoretic outer bounds that match the achievable regions for a certain range of parameters.

In the rest of the paper we introduce the system model and state the main results in Section II and the main results in Section III. We review single-stream codes in Section refsec:bckg. We provide the associated coding schemes and the converse bounds in Sections V-VII. Simulations results are presented in Section VIII and we conclude in Section IX.

A. Notation

Throughout this paper we use the following notation. A symbol over the field $\mathbb{F}_q$ is denoted using standard font, e.g., $a \in \mathbb{F}_q$. A vector of symbols is denoted using lower case bold font e.g., $\mathbf{s} \in \mathbb{F}_q^n$. The source packets at time t for streams a and b are denoted by $\mathbf{s}_a[t]$ and $\mathbf{s}_b[t]$ respectively while the channel packet is denoted using $\mathbf{x}[t]$. The notation $\mathbf{s}_a\begin{bmatrix} t \\ 0 \end{bmatrix}$ denotes the collection of source packets in the interval $[0,t]$. Matrices are denoted using upper case bold font, e.g. $\mathbf{P}$. The function $H(\cdot)$ denotes the entropy function and $I(\cdot;\cdot)$ the mutual information between two random variables.

II. System Model

Consider that we have two source streams $\{\mathbf{s}_a[t]\}_{t\geq 0}$ and $\{\mathbf{s}_b[t]\}_{t\geq 0}$. Assume that each source-packet in stream a, $\mathbf{s}_a[t] \in \mathbb{F}_q^{k_a}$ and likewise in stream b, $\mathbf{s}_b[t] \in \mathbb{F}_q^{k_b}$, where $\mathbb{F}_q$ denotes the underlying field for the source symbols. We assume that for each time $t \geq 0$, the source-packets $\mathbf{s}_a[t]$ and $\mathbf{s}_b[t]$ are revealed to the encoder at time t. The encoder generates a channel-packet $\mathbf{x}[t] \in \mathbb{F}_q^n$ that can depend on all the past source-packets, i.e.,

$$\mathbf{x}[t] = f_t\left(\mathbf{s}_a\begin{bmatrix} t \\ 0 \end{bmatrix}, \mathbf{s}_b\begin{bmatrix} t \\ 0 \end{bmatrix}\right),\ t \geq 0,$$

where recall that $\mathbf{s}_a\begin{bmatrix} t_2 \\ t_1 \end{bmatrix}$ refers to the collection of packets $\{\mathbf{s}_a[t_1], \ldots, s_a[t_2]\}$ etc. and $f_t(\cdot)$ denotes the encoding function at time t. The channel is a burst erasure channel, i.e., starting at some arbitrary time j, it erases at-most B consecutive packets. The output $\mathbf{y}[t]$ is defined as

$$\mathbf{y}[t] = \begin{cases} \star, & t \in [j, j+B'-1] \\ \mathbf{x}[t], & \text{otherwise} \end{cases}$$

for some integer $j \geq 0$ and $B' \leq B$.

Finally, we specify the decoding of the source-packets $s_a[t]$ and $s_b[t]$. Source stream a has the decoding delay T_a and source stream b has the decoding delay T_b,

$$\hat{\mathbf{s}}_a[t] = g_{a,t}\left(\mathbf{y}\begin{bmatrix} t+T_a \\ 0 \end{bmatrix}\right),\quad \hat{\mathbf{s}}_b[t] = g_{b,t}\left(\mathbf{y}\begin{bmatrix} t+T_b \\ 0 \end{bmatrix}\right) \qquad (1)$$

where $g_{a,t}(\cdot)$ and $g_{b,t}(\cdot)$ denote the decoding functions associated with streams a and b respectively.

A (B, T_a, T_b) code is able to recover each source-packet $s_a[t]$ with a delay of T_a and each source-packet $s_b[t]$ with a delay of T_b, i.e., we require that $\hat{\mathbf{s}}_a[t] = \mathbf{s}_a[t]$ and $\hat{\mathbf{s}}_b[t] = \mathbf{s}_b[t]$ for all $t \geq 0$. The rate of the code will be characterized by an ordered pair which has the rate for source a and the rate for source b.

$$(R_a, R_b) = \left(\frac{k_a}{n}, \frac{k_b}{n}\right) \qquad (2)$$

A rate pair (R_a, R_b) is achievable if there exist a (B, T_a, T_b) code over some field $\mathbb{F}_q$ such that k_a, k_b, n satisfy (2). The capacity region is the convex hull of all achievable rate pairs (R_a, R_b).

Throughout, without loss of generality, assume that $B \leq T_b < T_a$. Stream "b" is the *urgent* stream whereas stream "a" as the *less urgent* stream. All our construction will be systematic codes where the channel packet at time t will include both the source packets $\mathbf{s}_a[t]$ and $\mathbf{s}_b[t]$ i.e.,

$$\mathbf{x}[t] = \begin{pmatrix} \mathbf{s}_a[t] \\ \mathbf{s}_b[t] \\ \mathbf{p}[t] \end{pmatrix} \qquad (3)$$

where $\mathbf{p}[t]$ is the parity check packet at time t consisting of $n - (k_a + k_b)$ symbols. Such constructions are practically important as each received channel packet guarantees the immediate recovery of the associated source packets.

Remark 1: Note that our channel introduces a single erasure burst of maximum length B. It can be easily seen that all our constructions immediately apply to a channel that introduces multiple erasure bursts, each of length at-most B and having a guard separation of at-least T_a where there are no erasures. This follows since each burst is fully recovered with a delay of T_a and thus the system will reset at this point and can handle a fresh erasure burst. Furthermore in our simulations we will consider Gilbert channels, where both the erasure burst length, and gaps between successive bursts are not fixed, but random variables.

III. Main Results

The analysis of the capacity region involves three different regimes based on the delay constraint of the less urgent stream with delay T_a as discussed below.

A. Large Delay Regime: $T_a \geq T_b + 2B$

Theorem 1: An achievable rate region when $T_a \geq T_b + 2B$ consists of all rate pairs satisfying: $R_a, R_b \geq 0$ and

$$\left(\frac{T_b}{T_b+B}\right) R_a + R_b \leq \frac{T_b}{T_b+B}, \tag{4a}$$

$$R_a + R_b \leq \frac{T_a}{T_a+B}. \tag{4b}$$

Furthermore the above region is the capacity region in the following special cases (i) the delay of urgent stream is minimum possible, i.e., $T_b = B$, (ii) the class of systematic codes, see Eq. (3), and (iii) when $\frac{T_a - T_b}{T_a+B} \leq R_a \leq \frac{T_a}{T_a+B}$ is satisfied. □

Fig. 2a illustrates the region associated with Theorem 1. Note that the region has three corner points. The point $R_a = \frac{T_a}{T_a+B}$ and $R_b = 0$ corresponds to the single flow capacity [1], [23] when only flow a is present, whereas the point $R_a = 0$ and $R_b = \frac{T_b}{T_b+B}$ corresponds to the other extreme case when only flow b is present. The third corner point is $R_a = \frac{T_a - T_b}{T_a+B}$ and $R_b = \frac{T_b}{T_b+B}$. At this point both flows simultaneously co-exist, even though the sum-rate equals $\frac{T_a}{T_a+B}$, i.e., the capacity associated with stream a only. We will provide a code construction that achieves this corner point and then explain how the rest of the capacity region can be obtained using the other two extreme points. Note that in the range $\frac{T_a - T_b}{T_a+B} \leq R_a \leq \frac{T_a}{T_a+B}$, the sum rate constraint (4b) is active and the proposed coding scheme is optimal. We will establish the upper bound (4a) for the case of systematic codes and also for non-systematic codes in the special case when $T_b = B$.

B. Low Delay Regime: $T_b < T_a \leq T_b + B$

We next consider the case when the less urgent stream has a delay $T_a \in (T_b, T_b + B]$. In this case we have been able to find a complete characterization of the capacity region.

Theorem 2: The capacity region for a system with $T_b < T_a \leq T_b + B$ is given by the set of all rate pairs that satisfy: $R_a, R_b \geq 0$ and

$$\left(\frac{2T_b + B - T_a}{T_b+B}\right) R_a + R_b \leq \frac{T_b}{T_b+B}, \tag{5a}$$

$$R_a + R_b \leq \frac{T_a}{T_a+B}. \tag{5b}$$

□

Fig. 2b illustrates the tradeoff between R_a and R_b. Again note that the region has three corner points. The points $R_a = \frac{T_a}{T_a+B}$, $R_b = 0$ and $R_a = 0$, $R_b = \frac{T_b}{T_b+B}$, correspond to single-flow capacities. The point $R_a = \frac{B}{T_a+B}$ and $R_b = \frac{T_a - B}{T_a+B}$ corresponds to the case when both flows simultaneously exist, yet sum rate is given by the capacity of flow a only. We will propose a code construction that achieves this corner point, and explain how the rest of the region can be achieved by "time-sharing" with the other two extreme points. Also note that whenever $\frac{B}{T_a+B} \leq R_a \leq \frac{T_a}{T_a+B}$, holds the sum-rate constraint (5b) will be active and our proposed scheme is clearly optimal. Thus we will provide a converse for upper bound associated with (5a) for the general case of non-systematic codes.

C. Intermediate Delay Regime: $T_b + B < T_a < T_b + 2B$

In the intermediate delay regime we have that the following rate region is achievable.

Theorem 3: When $T_b + B < T_a < T_b + 2B$, all rate pairs that satisfy $R_a \geq 0, R_b \geq 0$,

$$R_a + R_b \leq \frac{T_a}{T_a+B}, \tag{6}$$

$$\left(\frac{T_b}{T_b+B}\right) R_a + R_b \leq \frac{T_b}{T_b+B}, \tag{7}$$

$$\begin{aligned}(4B + 3T_b - 2T_a)R_a &+ (3B + 2T_b - T_a)R_b \\ &\leq (2B + 2T_b - T_a)\end{aligned} \tag{8}$$

are achievable. Furthermore for the case of systematic codes (3) any achievable rate pair satisfies $R_a \geq 0, R_b \geq 0$ as well as the conditions in (4a) and (4b). □

Fig. 2c illustrates the achievable region and the outer bound stated in Theorem 3. In the inner-bound we have two corner points: $(R_a, R_b) = \left(\frac{B}{T_b+2B}, \frac{T_b}{T_b+2B}\right)$ and $(R_a, R_b) = \left(\frac{2B}{T_a+B}, \frac{T_a - 2B}{T_a+B}\right)$. The upper bound for systematic codes meets the inner bound for the entire region except the segment joining these two corner points (shown by the red-line).

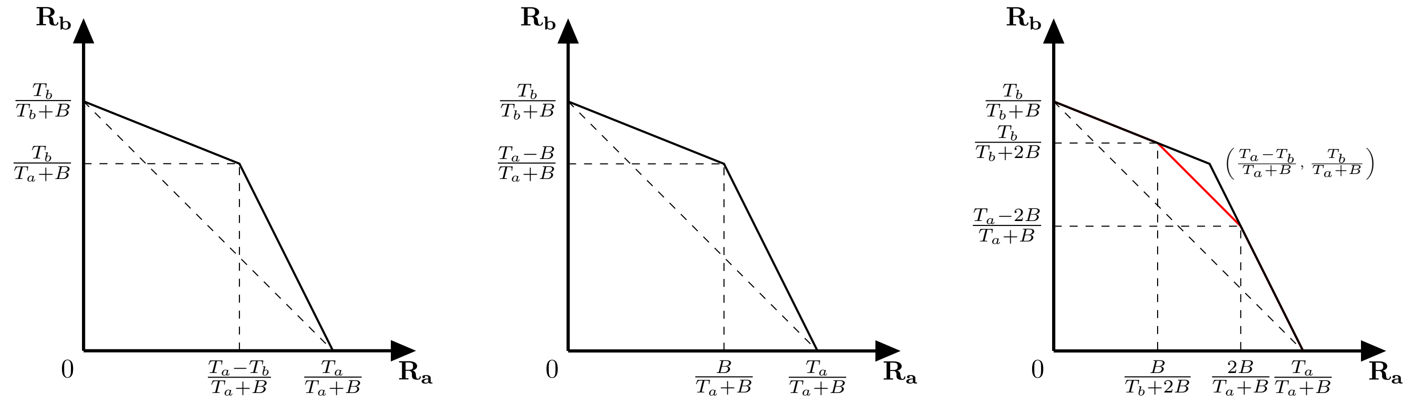
Before presenting the proofs we discuss the following example.

D. Example: $T_b = B, T_a = 2B$

Our code construction simplifies considerably in the special case when $T_a = T = 2B$ and $T_b = B$ holds. It follows from Theorem 2 that $R_a = R_b = \frac{1}{3}$ is a point on the boundary of the capacity region. The optimality is immediate since the sum rate $R_a + R_b = \frac{2}{3}$, corresponds to the single-flow capacity with the delay of $T = 2B$. For the achievability, assume that both $s_a[t], s_b[t] \in \mathbb{F}_q^k$ and let:

$$\mathbf{x}[t] = \begin{bmatrix} \mathbf{s}_a[t] \\ \mathbf{s}_b[t] \\ \mathbf{s}_a[t-2B] + \mathbf{s}_b[t-B] \end{bmatrix} \in \mathbb{F}_q^{3k}. \tag{9}$$

It follows that this construction satisfies $R_a = R_b = \frac{1}{3}$. To see the feasibility, we need to show that when the channel introduces an erasure burst of length B, the delays associated with streams a and b are given by $T_a = 2B$ and $T_b = B$ respectively. Suppose without loss of generality that the erasure burst spans the interval $t \in [i, i+B-1]$. Note that since the construction in (9) is systematic, all the source-packets



(a) Achievable Region: $T_a \geq T_b + 2B$. (b) Capacity when $T_b < T_a < T_b + B$. (c) Bounds on capacity of (B, T, B) code where $2B < T < 3B$.

Fig. 2: The solid line shows the rate region achievable with joint coding scheme for the three cases. A sub-optimal scheme that encodes the two source streams separately using the constructions in [1], [23], and concatenates them, achieves the straight line connecting $\left(0, \frac{T_a}{T_a+B}\right)$ and $\left(0, \frac{T_b}{T_b+B}\right)$. This is shown by the dashed line connecting these points.

before time $t = i$ are available to the decoder and likewise all source-packets from time $t = i + B$ are also available immediately. We need to show that for $t \in [i, i+B-1]$ each source-packet $\mathbf{s}_a[t]$ is recovered with a delay of $T_a = 2B$ and each source-packet $\mathbf{s}_b[t]$ is recovered with a delay of $T_b = B$ respectively. In particular let $t' = t + B$. The parity check at time t' can be expressed as $\mathbf{p}[t'] = \mathbf{s}_a[t-B] + \mathbf{s}_b[t]$. Since $t - B < i$ it follows that $\mathbf{s}_a[t]$ is available to the decoder. Hence it can be cancelled and the remaining source-packet $\mathbf{s}_b[t]$ is recovered at time $t' = t + B$, i.e., with a delay of B. Since the above argument holds for each $t \in [i, i+B-1]$ it follows that each erased source-packet $\mathbf{s}_b[t]$ is recovered with a delay of B. Next consider $t'' = t + 2B$ and observe that the parity check, $\mathbf{p}[t''] = \mathbf{s}_a[t] + \mathbf{s}_b[t+B]$. Since $t+B > i+B-1$ it follows that $\mathbf{s}_b[t]$ is available to the decoder. Hence it can be cancelled and the remaining source-packet $\mathbf{s}_a[t]$ is recovered at time $t'' = t + 2B$, i.e., with a delay of $2B$ as required.

For general values of T, while the basic idea of recovering the source-packets of $\mathbf{s}_b[t]$ first is still used, the underlying constructions are based on a more complex class of streaming codes, as discussed next.

IV. Background

We discuss two schemes from prior works for the single stream setting. Our constructions for the multi-stream setting will build upon these codes.

A. Random Linear Codes

A systematic Random Linear Code (RLC) maps the stream of source packets $\mathbf{s}[t] \in \mathbb{F}_q^k$ into a stream of channel packets $\mathbf{x}[t] \in \mathbb{F}_q^n$ where $\mathbf{x}[t] = (\mathbf{s}[t], \mathbf{p}[t])$. The parity check packet $\mathbf{p}[t] \in \mathbb{F}_q^{n-k}$ is generated as a linear combination of the past M source packets, where M denotes the memory of the code:

$$\mathbf{p}[t] = \sum_{j=1}^{M} \mathbf{s}[t-j] \cdot \mathbf{P}_j, \tag{10}$$

where the matrices $\mathbf{P}_j$ are $k \times (n-k)$ parity matrices whose entries are selected at random (see e.g., [24], [25]) from a sufficiently large field to guarantee linear independence of the parity check equations. Deterministic variants of these codes are also known, see e.g., [26].

In the streaming setting following an erasure burst of length B, each source packet can be recovered with a delay of T using a RLC with memory $M \geq T$, provided that:

$$B \leq (1-R)(T+1) \tag{11}$$

where $R = k/n$ is the rate of RLC. We note that the right hand side in (11) also corresponds to the maximum possible column distance of a convolutional code of rate R, see [2].

B. Maximally-Short (MS) Codes

A (B, T) MS code is a rate $R = \frac{T}{T+B}$ convolutional code that can recover from a single erasure burst of maximum length B within a delay of $\max(B, T)$. MS codes were introduced in [1], [23]. The construction described here is based on the layered coding approach in [2].

1) Code Construction: We assume that each source packet consists of k symbols and split it into sub-packets:

$$\mathbf{s}[t] = (\mathbf{u}[t], \mathbf{v}[t]), \tag{12}$$

where $\mathbf{u}[t] \in \mathbb{F}_q^{\frac{B}{T}k}$ and $\mathbf{v}[t] \in \mathbb{F}_q^{\frac{T-B}{T}k}$. The parity-check packet at time t is given by,

$$\mathbf{p}[t] = \mathbf{u}[t-T] + \sum_{j=1}^{T} \mathbf{v}[t-j]\mathbf{P}_j, \tag{13}$$

where $\mathbf{P}_j \in \mathbb{F}_q^{\frac{T-B}{T}k \times \frac{B}{T}k}$ are the matrices of a RLC, see Eq. (10). Hence the parity-check packets $\mathbf{p}[t] \in \mathbb{F}_q^{\frac{B}{T}k}$ and the channel packet is $\mathbf{x}[t] = (\mathbf{s}[t], \mathbf{p}[t])$. The rate of the code is $R = \frac{T}{T+B}$.

2) Decoding: Suppose that the erasure burst spans the interval $[i, i+B-1]$. Since the code is systematic the decoder has all the source packets before time i and after time $i+B-1$ immediately available. We need to show that every source packet $\mathbf{s}[t] = (\mathbf{u}[t], \mathbf{v}[t])$ for $t \in [i, i+B-1]$ is recovered by time $t+T$. Towards that end we will show that (i) all the source sub-packets $\mathbf{v}[i], \ldots, \mathbf{v}[i+B-1]$ are recovered simultaneously at time $i+T-1$ and (ii) each $\mathbf{u}[t]$ is recovered at time $t+T$.

First note that in the interval $t \in [i+B, i+T-1]$ the parity-check packets $\mathbf{p}[t] = \mathbf{u}[t-T] + \sum_{j=0}^{T} \mathbf{v}[t-j]\mathbf{P}_j$ consist of two parts. The contribution of $\mathbf{u}[t-T]$, which is available to the decoder as $t-T < i$ and the contribution $\mathbf{q}[t] = \sum_{j=0}^{T} \mathbf{v}[t-j]\mathbf{P}_j$, which is a random linear code applied to the $\mathbf{v}[\cdot]$ packets of rate $R' = \frac{T-B}{T}$. Thus the decoder can recover $\mathbf{q}[t]$ for $t \in [i+B, i+T-1]$, yielding a total of $(T-B)\frac{Bk}{T}$ equations. These suffice to recover all the erased $\mathbf{v}[\cdot]$ packets in the interval $[i, i+B-1]$ consisting of $(T-B)\frac{Bk}{T}$ symbols. The expression in (11) with a delay of $T-1$ and rate $R = \frac{T-B}{T}$ can be used to show the recovery as well.

By time $i+T-1$ all the packets in the $\mathbf{v}[\cdot]$ have been recovered. In the interval $[i+T, i+T+B-1]$ we can cancel the contribution of $\sum_{j=0}^{T} \mathbf{v}[t-j]\mathbf{P}_j$ from $\mathbf{p}[t]$ and recover $\mathbf{u}[t-T]$. At this point the entire source packet $\mathbf{s}[t-T]$ is recovered by time t i.e., with a delay of T as required.

C. MS Codes for Multiple Streams

For the multi-stream setting a straightforward approach for constructing a (B, T_a, T_b) multi-stream code is to apply a (B, T_a) MS code to stream a and a (B, T_b) MS code to stream b and then to concatenate the resulting parity check packets. Let us denote the resulting parity check packets from flows a and b by $\mathbf{p}_a[\cdot]$ and $\mathbf{p}_b[\cdot]$, respectively. The overall channel packet is given by,

$$\mathbf{x}[t] = \begin{pmatrix} \mathbf{s}_a[t] \\ \mathbf{s}_b[t] \\ \mathbf{p}_a[t] \\ \mathbf{p}_b[t] \end{pmatrix} \tag{14}$$

In order to compute the associated rate region, let us assume that $\mathbf{s}_a[t] \in \mathbb{F}_q^{k_a}$ and $\mathbf{s}_b[t] \in \mathbb{F}_q^{k_b}$. From the construction of the MS codes, note that $\mathbf{p}_a[t] \in \mathbb{F}_q^{k_a \frac{B}{T_a}}$ and $\mathbf{p}_b[t] \in \mathbb{F}_q^{k_b \frac{B}{T_b}}$. Using the fact that $n = k_a\left(1 + \frac{B}{T_a}\right) + k_b\left(1 + \frac{B}{T_b}\right)$. It follows that the rate region that is achieved is given by:

$$\frac{T_a + B}{T_a} R_a + \frac{T_b + B}{T_b} R_b \leq 1 \tag{15}$$

Remark 2: The rate region in (15) is just a straight line connecting the extreme points $\left(\frac{T_a}{T_a+B}, 0\right)$ and $\left(0, \frac{T_b}{T_b+B}\right)$. We will hence refer to it as *time-sharing* scheme. In our subsequent constructions, when showing the achievability of a rate pair on a straight-line connecting two corner points, we will invoke such a time-sharing approach.

V. Proof of Theorem 1

We provide a proof of Theorem 1 by presenting an achievability for $T_a \geq T_b + 2B$ and later prove the optimality for the special cases listed in Theorem 1.

As noted in Fig. 2c, the boundary of the capacity region is a piecewise linear function. It suffices to show that the point

$$\bar{R}_a = \frac{T_a - T_b}{T_a + B}, \tag{16a}$$

$$\bar{R}_b = \frac{T_b}{T_a + B} \tag{16b}$$

is achievable. The rest of the boundary can be traced by "time-sharing," i.e., splitting each source-packet into two parts, applying a single-flow code achieving either $(R_a = \frac{T_a}{T_a+B}, R_b = 0)$ or $(R_a = 0, R_b = \frac{T_b}{T_b+B})$ in one part, and the code achieving $(\bar{R}_a, \bar{R}_b)$ in the other part; see remark 2.

We now provide a detailed construction for the point $(\bar{R}_a, \bar{R}_b)$. The high level idea in the construction is to apply MS codes to both stream a and b. We apply a (B, T_b) MS code to stream b and a $(B, T_a - T_b - B)$ MS code to stream a. We shift the parity checks of stream a by $T_b + B$ to avoid interference with stream b. This allows us to recover the source packets in stream b first with a delay of T_b. It turns out that the shift of $T_b + B$ for stream a, also allows us to embed an additional set of source packets in stream a using a repetition code. This is described in detail below.

- Let $\mathbf{s}_a[t] \in \mathbb{F}_q^{\frac{T_a - T_b}{T_a}k}$ and split it into $\mathbf{s}_a[t] = (\mathbf{s}_{a,1}[t], \mathbf{s}_{a,2}[t])$, where $\mathbf{s}_{a,1}[t] \in \mathbb{F}_q^{\frac{T_a - T_b - B}{T_a}k}$, $\mathbf{s}_{a,2}[t] \in \mathbb{F}_q^{\frac{B}{T_a}k}$. The sub-packet $\mathbf{s}_{a,1}[t]$ will be protected using a MS code, while $\mathbf{s}_{a,2}[t]$ will be protected using a repetition code.
- Apply a $(B, T_a - T_b - B)$ MS code to $\mathbf{s}_{a,1}[t] = (\mathbf{u}_{a,1}[t], \mathbf{v}_{a,1}[t])$ and shift the resulting parities by $T_b + B$, i.e.,

$$\mathbf{p}_{a,1}[t] = \mathbf{u}_{a,1}[t - T_a] + \sum_{j=T_b+B}^{T_a} \mathbf{v}_{a,1}[t-j] \cdot \mathbf{P}_{a,j}, \tag{17}$$

  where $\mathbf{u}_{a,1}[t] \in \mathbb{F}_q^{\frac{B}{T_a}k}$, $\mathbf{v}_{a,1}[t] \in \mathbb{F}_q^{\frac{T_a - T_b - 2B}{T_a}k}$ and $\mathbf{P}_{a,j} \in \mathbb{F}_q^{\frac{T_a - T_b - 2B}{T_a}k \times \frac{B}{T_a}k}$. Recall that $T_a \geq T_b + 2B$ guarantees that such a construction is feasible.
- Apply a $T_b + B$ shifted repetition code to $\mathbf{s}_{a,2}[\cdot]$ generating,

$$\mathbf{p}_{a,2}[t] = \mathbf{s}_{a,2}[t - T_b - B]. \tag{18}$$

- Let $\mathbf{s}_b[t] \in \mathbb{F}_q^{\frac{T_b}{T_a}k}$ and apply a (B, T_b) MS code to $\mathbf{s}_b[t]$ to generate the parity-check packets,

$$\mathbf{p}_b[t] = \mathbf{u}_b[t - T_b] + \sum_{j=0}^{T_b} \mathbf{v}_b[t-j] \cdot \mathbf{P}_{b,j}, \tag{19}$$

  where $\mathbf{u}_b[t] \in \mathbb{F}_q^{\frac{B}{T_a}}$ and $\mathbf{v}_b[t] \in \mathbb{F}_q^{\frac{T_b - B}{T_a}}$, $\mathbf{s}_b[t] = (\mathbf{u}_b[t], \mathbf{v}_b[t])$ and $\mathbf{P}_{b,j} \in \mathbb{F}_q^{\frac{T_b - B}{T_a}k \times \frac{B}{T_a}k}$.
- The channel packet at time t is

$$\mathbf{x}[t] = \begin{pmatrix} \mathbf{s}_a[t] \\ \mathbf{s}_b[t] \\ \mathbf{p}[t] = \mathbf{p}_a[t] + \mathbf{p}_b[t] \end{pmatrix} \tag{20}$$

  where $\mathbf{p}_a[t] = \mathbf{p}_{a,1}[t] + \mathbf{p}_{a,2}[t]$.

The code in (20) achieves the rate-pair in (16).

To demonstrate the feasibility, assume that an erasure burst spans the interval $[i, i+B-1]$. Since the construction in (20) is a systematic code, the source-packets $\mathbf{s}_a[t]$ and $\mathbf{s}_b[t]$ for $t \notin [i, i+B-1]$ are immediately available to the decoder. The decoder proceeds as follows to recover the erased source packets.

(1) Recover $\mathbf{s}_b[i], \ldots, \mathbf{s}_b[i+B-1]$. Let $t \in [i+B, i+T_b+B-1]$:

- According to (17), the shift of $T_b + B$ guarantees that the parities $\mathbf{p}_{a,1}[t]$ combine $\mathbf{s}_{a,1}[\cdot]$ from time $i-1$ and earlier, which are not erased. Also, the parities $\mathbf{p}_{a,2}[t] = \mathbf{s}_{a,2}[t - T_b - B]$ are before the erasure burst, and are not erased. Therefore, both $\mathbf{p}_{a,1}[t]$ and $\mathbf{p}_{a,2}[t]$ can be computed and subtracted from $\mathbf{p}[t]$ to recover $\mathbf{p}_b[t]$.
- The recovered parities, $\mathbf{p}_b[t]$, are (B, T_b) MS code parities that can recover $\mathbf{s}_b[i], \ldots, \mathbf{s}_b[i+B-1]$ with a delay of T_b.

(2) Recover $\mathbf{v}_{a,1}[i], \ldots, \mathbf{v}_{a,1}[i+B-1]$. Let $t \in [i+T_b+2B, i+T_a-1]$:

- All $\mathbf{s}_b[\cdot]$ packets are recovered in the previous step and hence $\mathbf{p}_b[t]$ can be computed and subtracted to recover $\mathbf{p}_a[t]$. Furthermore, the parities $\mathbf{p}_{a,2}[t]$ combine $\mathbf{s}_{a,2}[t - T_b - B]$ which are after the erasure burst and hence can be subtracted from $\mathbf{p}_a[t]$ to recover $\mathbf{p}_{a,1}[t]$.
- The $\mathbf{u}_{a,1}[t]$ are from before the burst and can also be subtracted from $\mathbf{p}[t]$ leaving out $\mathbf{p}_{a,2}[t] = \mathbf{s}_{a,2}[t - T_a]$ for $t \in [i+T_b+2B, i+T_a-1]$. This gives $(T_a - T_b - 2B)B\frac{k}{T_a}$ equations which suffice to recover $\mathbf{v}_{a,1}[i], \ldots, \mathbf{v}_{a,1}[i+B-1]$ at time $i+T_a-1$, i.e., with a maximum delay of $T_a - 1$.

(3) Recover $\mathbf{s}_{a,2}[i], \ldots, \mathbf{s}_{a,2}[i+B-1]$. Let $t \in [i+T_b+B, i+T_b+2B-1]$:

- Since all erased $\mathbf{s}_b[\cdot]$ and $\mathbf{v}_{a,1}[\cdot]$ packets are decoded at time $i+T_b+B-1$ and $i+T_a-1$ in the previous steps, their effect can be subtracted from $\mathbf{p}[t]$ to recover $\mathbf{p}_{a,2}[t] + \mathbf{u}_{a,1}[t - T_a]$.
- The $\mathbf{u}_{a,1}[t - T_a]$ are from before the erasure burst and hence can also be subtracted to recover $\mathbf{s}_{a,2}[i], \ldots, \mathbf{s}_{a,2}[i+B-1]$ can be recovered.

(4) Recover $\mathbf{u}_{a,1}[i], \ldots, \mathbf{u}_{a,1}[i+B-1]$. Let $t \in [i+T_a, i+T_a+B-1]$:

- Since all erased $\mathbf{s}_b[\cdot]$, $\mathbf{v}_{a,1}[\cdot]$ and $\mathbf{s}_{a,2}[\cdot]$ packets are recovered by time $i+T_a-1$, the decoder can subtract their effect from the $\mathbf{p}[t]$ to recover $\mathbf{u}_{a,1}[i], \ldots, \mathbf{u}_{a,1}[i+B-1]$ with a delay of T_a.

At this point the source packet $\mathbf{s}_a[t] = (\mathbf{u}_{a,1}[t], \mathbf{v}_{a,1}[t], \mathbf{s}_{a,2}[t])$ is recovered with a delay of T_a.

A. Converse: Systematic Codes

Note that the inequality (4b) follows by relaxing the delay constraint of flow b to $T_b = T_a$. It follows from the single flow capacity [1] that $R_a + R_b \leq \frac{T_a}{T_a+B}$ must hold for any code, systematic or non-systematic.

In the case of systematic codes (3), the constraint (4a) can be established by the following argument. Suppose the erasure burst spans the interval $[0, B-1]$ and assume that T_b symbols following it are not erased. Since the code is systematic we

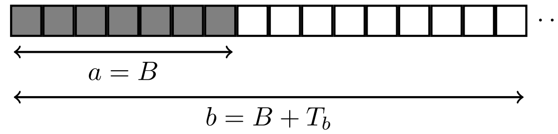


Fig. 3: The erasure channel used to establish (4a) for systematic codes. Grey and white squares denote erased and unerased symbols respectively.

have that the following conditions hold:

$$H\left(\mathbf{s}_a\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right] \middle| \mathbf{x}\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right]\right) = 0 \tag{21}$$

$$H\left(\mathbf{s}_b\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right] \middle| \mathbf{x}\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right]\right) = 0 \tag{22}$$

Furthermore since each source packet in $\mathbf{s}_b[\cdot]$ must be recovered with a delay of T_b it follows that:

$$H\left(\mathbf{s}_b\left[\begin{smallmatrix} B-1 \\ 0 \end{smallmatrix}\right] \middle| \mathbf{x}\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right]\right) = 0 \tag{23}$$

Thus we have that:

$$T_b H(\mathbf{x}) \geq H\left(\mathbf{x}\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right]\right) \tag{24}$$

$$= H\left(\mathbf{s}_a\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right], \mathbf{s}_b\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right], \mathbf{s}_b\left[\begin{smallmatrix} B-1 \\ 0 \end{smallmatrix}\right], \mathbf{x}\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right]\right) \tag{25}$$

$$\geq H\left(\mathbf{s}_a\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right], \mathbf{s}_b\left[\begin{smallmatrix} T_b+B-1 \\ B \end{smallmatrix}\right], s_b\left[\begin{smallmatrix} B-1 \\ 0 \end{smallmatrix}\right]\right) \tag{26}$$

$$= T_b H(\mathbf{s}_a) + (T_b + B) H(\mathbf{s}_b) \tag{27}$$

where (25) follows from (21),(22) and (23). Thus we have that:

$$\frac{T_b}{T_b + B} H(\mathbf{x}) \geq \frac{T_b}{T_b + B} \frac{H(\mathbf{s}_a)}{H(x)} + H(\mathbf{s}_b) \tag{28}$$

Since $H(\mathbf{s}_a) = k_a$ and $H(\mathbf{s}_b) = k_b$ and $n \geq H(\mathbf{x})$, dividing throughout by n we recover (4a).

Remark 3: The derivation of upper bounds (4a) and (4b) does not make any assumption on T_a and applies to all the three cases for the case of systematic codes.

B. Converse: Non-Systematic Codes, $T_b = B$

For non-systematic codes, we focus on the special case of $T_b = B$. One can show that for any $r \geq T_a$ we must have that:

$$\begin{aligned}
&\sum_{i=0}^{r-1} H(\mathbf{x}[i]) \\
&\geq H\left(\mathbf{s}_a\left[\begin{smallmatrix} r-T_a-1 \\ 0 \end{smallmatrix}\right]\right) + H\left(\mathbf{s}_b\left[\begin{smallmatrix} r-B-1 \\ T_a-B \end{smallmatrix}\right]\right) + H\left(\mathbf{s}_b\left[\begin{smallmatrix} r-1 \\ T_a \end{smallmatrix}\right]\right) \\
&\quad + H\left(\mathbf{x}\left[\begin{smallmatrix} r-1 \\ 0 \end{smallmatrix}\right] \middle| \mathbf{s}_a\left[\begin{smallmatrix} r-T_a-1 \\ 0 \end{smallmatrix}\right] \mathbf{s}_b\left[\begin{smallmatrix} r-B-1 \\ T_a-B \end{smallmatrix}\right] \mathbf{s}_b\left[\begin{smallmatrix} r-1 \\ T_a \end{smallmatrix}\right]\right).
\end{aligned} \tag{29}$$

As a consequence of (29) we have that:

$$\begin{aligned}
\sum_{i=0}^{r-1} H(\mathbf{x}[i]) \geq\, & H\left(\mathbf{s}_a\left[\begin{smallmatrix} r-T_a-1 \\ 0 \end{smallmatrix}\right]\right) + H\left(\mathbf{s}_b\left[\begin{smallmatrix} r-B-1 \\ T_a-B \end{smallmatrix}\right]\right) \\
& + H\left(\mathbf{s}_b\left[\begin{smallmatrix} r-1 \\ T_a \end{smallmatrix}\right]\right)
\end{aligned} \tag{30}$$

$$r \cdot H(\mathbf{x}) \geq (r - T_a) \cdot H(\mathbf{s}_a) + 2 \cdot (r - T_a) \cdot H(\mathbf{s}_b)$$

$$\frac{r}{r - T_a} \geq \frac{H(\mathbf{s}_a)}{H(x)} + \frac{2H(\mathbf{s}_b)}{H(x)}. \tag{31}$$

Finally, this gives us

$$R_a + 2R_b \leq \frac{r}{r - T_a} \xrightarrow{r\to\infty} 1, \tag{32}$$

which is equivalent to (4a) at $T_b = B$.

The proof of (29) is established using mathematical induction and is relegated to Appendix I.

VI. Proof of Theorem 2

Theorem 2 considers the case when $T_b \leq T_a \leq T_b + B$. Its proof consists of two parts, an achievability scheme and a converse proof.

A. Achievability

From Fig. 2b it suffices to show that the following corner point is achievable:

$$\bar{R}_a = \frac{B}{T_a + B}, \tag{33a}$$

$$\bar{R}_b = \frac{T_a - B}{T_a + B}. \tag{33b}$$

The rest of the capacity region consists of straight lines connecting this corner point with the single-stream rates: $\left(\frac{T_a}{T_a+B}, 0\right)$ and $\left(0, \frac{T_b}{T_b+B}\right)$ as shown in Fig. 2b. These points can be attained by a time-sharing argument as indicated in remark 2.

Our code that achieves the rate pair $(\bar{R}_a, \bar{R}_b)$ in (33) is described next. The high level idea is to protect stream a using a repetition code, so that its interfering parity packets start appearing at time $i + T_a$ and later. This leaves a window of $[i + B, i + T_a - 1]$ to decode stream b. We accomplish this as follows. We first assume that $T_a \geq 2B$:

- Let $\mathbf{s}_a[t] \in \mathbb{F}_q^{\frac{B}{T_a}k}$ and apply a repetition code with a shift of T_a,
$$\mathbf{p}_a[t] = \mathbf{s}_a[t - T_a] \in \mathbb{F}_q^{\frac{B}{T_a}k}. \tag{34}$$
- Let $\mathbf{s}_b[t] \in \mathbb{F}_q^{\frac{T_a-B}{T_a}k}$ and apply a $(B, T_a - B)$ MS code to $\mathbf{s}_b[t]$ to generate the parity-check packets,
$$\mathbf{p}_b[t] = \mathbf{u}_b[t - T_a + B] + \sum_{j=0}^{T_a-B} \mathbf{v}_b[t - j]\mathbf{P}_{b,j}, \tag{35}$$
where $\mathbf{u}_b[t] \in \mathbb{F}_q^{\frac{B}{T_a}k}$, $\mathbf{v}_b[t] \in \mathbb{F}_q^{\frac{T_a-2B}{T_a}k}$, $\mathbf{s}_b[t] = (\mathbf{u}_b[t], \mathbf{v}_b[t])$, $\mathbf{P}_{b,j} \in \mathbb{F}_q^{\frac{T_a-2B}{T_a}k\times\frac{B}{T_a}k}$ are the matrices in the MS code. Note that $\mathbf{p}_b[t] \in \mathbb{F}_q^{\frac{B}{T_a}k}$.
- The channel packet at time t is
$$\mathbf{x}[t] = \begin{pmatrix} \mathbf{s}_a[t] \\ \mathbf{s}_b[t] \\ \mathbf{p}[t] = \mathbf{p}_a[t] + \mathbf{p}_b[t] \end{pmatrix}. \tag{36}$$

It is clear that $\mathbf{x}[t] \in \mathbb{F}_q^{\frac{T_a+B}{T_a}k}$ and the rate pair achieved is $(\bar{R}_a, \bar{R}_b)$ in (33).

To show that the proposed construction is feasible suppose that an erasure burst spans the interval $[i, i + B - 1]$. Since the code in (36) is systematic, all source packets, $\mathbf{s}_a[t]$ and $\mathbf{s}_b[t]$, outside the erased interval are available at the decoder. The decoder recovers the erased source packets in the interval $[i, i + B - 1]$ as follows:

(1) Recover $\mathbf{s}_b[i], \ldots, \mathbf{s}_b[i+B-1]$: Let $t \in [i+B, i+T_a-1]$,

- The decoder first subtracts the parities $\mathbf{p}_a[t] = \mathbf{s}_a[t - T_a]$ which are not erased in the interval $[i + B, i + T_a - 1]$ to recover $\mathbf{p}_b[t]$.
- Since we apply a $(B, T_a - B)$ MS code, each source packet is recovered with a delay of $T_a - B$. Thus the last erased source packet $\mathbf{s}_b[i + B - 1]$ is recovered at time $i + T_a - 1$ as required.

(2) Recover $\mathbf{s}_a[i], \ldots, \mathbf{s}_a[i + B - 1]$: Let $t \in [i + T_a, i + T_a + B - 1]$,

- Since all $\mathbf{s}_b[\cdot]$ are recover by time $i + T_a - 1$ in the previous step, the decoder can compute the parities $\mathbf{p}_b[t]$ and subtract them to recover $\mathbf{p}_a[t]$.
- These parities combine repetitions of $\mathbf{s}_a[t - T_a]$ and hence $\mathbf{s}_a[i], \ldots, \mathbf{s}_a[i + B - 1]$ can be recovered with a delay of T_a.

In the case when $T_a < 2B$, one cannot use a $(T_a - B, B)$ MS code on stream b. Instead we use a combination of a repetition code and RLC as discussed next. In particular let

$$\mathbf{p}_{b,1}[t] = \mathbf{s}_b[t - B] \tag{37}$$

be a repetition code with shift B. Note that $\mathbf{p}_{b,1}[t] \in \mathbb{F}_q^{\frac{T_a-B}{T_a}k}$. Furthermore let

$$\mathbf{p}_{b,2}[t] = \sum_{j=0}^{T_a-1} \mathbf{s}_b[t - j]\mathbf{P}_{b,2,j} \tag{38}$$

where $\mathbf{P}_{b,2,j} \in \mathbb{F}_q^{\frac{T_a-B}{T_a}k\times\frac{2B-T_a}{T_a}k}$ are the encoding matrices of RLC (see section IV-A) with memory T_a. We let

$$\mathbf{p}_b[t] = \begin{pmatrix} \mathbf{p}_{b,1}[t] \\ \mathbf{p}_{b,2}[t] \end{pmatrix} \in \mathbb{F}_q^{\frac{B}{T_a}k}. \tag{39}$$

The parity checks $\mathbf{p}_a[t]$ for stream a are generated as before using shifted repetition code and the overall channel packet is as in (36).

To show the feasibility of the proposed construction we assume that an erasure burst spans the interval $[i, i + B - 1]$ and show that the source packets $\mathbf{s}_b[i], \ldots, \mathbf{s}_b[i + B - 1]$ can be recovered in the interval $[i + B, i + T_a - 1]$. Note that in this interval the parity $\mathbf{p}_a[t] = \mathbf{s}_a[t - T_a]$ are non-erased and can be cancelled out to recover $\mathbf{p}_b[t]$. The parity packets $\mathbf{p}_{b,1}[t]$ are repetition codes that can be used to directly recover $\mathbf{s}_b[i], \ldots, \mathbf{s}_b[[i+T_a-B-1]$ with a delay of B. The remaining source packets $\mathbf{s}_b[i+T_a-B], \ldots, \mathbf{s}_b[i+B-1]$ have a deadline after time $i + T_a - 1$ and can be simultaneously recovered at time $i + T_a - 1$ using $\mathbf{p}_{b,2}[i + B], \ldots, \mathbf{p}_{b,2}[i + T_a - 1]$. Note that each $\mathbf{p}_{b,2}[t] \in \mathbb{F}_q^{\frac{2B-T_a}{T_a}k}$ and thus we have a total of $\left(\frac{2B-T_a}{T_a}\right)(T_a - B)k$ equations from these parity-check packets. These equal the number of symbols in $\mathbf{s}_b[i + T_a -$

$B], \ldots, \mathbf{s}_b[i+B-1]$, as each source packet consists of $\frac{T_a-B}{T_a}k$ symbols over $\mathbb{F}_q$. Thus all the erased source symbols in $\mathbf{s}_b[\cdot]$ can be recovered by time $i+T_a-1$ as required[1].

B. Converse

We will establish the non-trivial bound (5a) in the converse. We first outline the converse argument for the case of systematic codes. Our proof is based on the erasure sequence in Figure 3. We consider $B+T_b$ channel packets with the first B of which are erased followed by T_b non-erasures in the interval $[0, T_b+B-1]$. The first B source packets of the first stream, $\mathbf{s}_b[0], \ldots, \mathbf{s}_b[B-1]$ can be decoded with a delay of T_b, i.e., by time T_b+B-1.

$$H\left(\mathbf{s}_b\begin{bmatrix}B-1\\0\end{bmatrix} \middle| \mathbf{x}\begin{bmatrix}T_b+B-1\\B\end{bmatrix}\right) = 0 \tag{40}$$

The source packets of the less urgent stream, $\mathbf{s}_a[\cdot]$, require a delay of T_a. Hence, only the first $B+T_b-T_a$ can be decoded by the end of the period, i.e., at T_b+B-1.

$$H\left(\mathbf{s}_a\begin{bmatrix}B+T_b-T_a-1\\0\end{bmatrix} \middle| x\begin{bmatrix}T_b+B-1\\B\end{bmatrix}\right) = 0 \tag{41}$$

In case of systematic codes, all source packets in the interval, $[B, T_b+B-1]$, can be directly decoded.

$$H\left(\mathbf{s}_a\begin{bmatrix}T_b+B-1\\B\end{bmatrix}, s_b\begin{bmatrix}T_b+B-1\\B\end{bmatrix} \middle| x\begin{bmatrix}T_b+B-1\\B\end{bmatrix}\right) = 0 \tag{42}$$

$$\tag{43}$$

By combining (40), (41) and (42) and along the lines of (28) we can establish that:

$$T_b H(\mathbf{x}) \geq (T_b+B)H(\mathbf{s}_b) + (B+2T_b-T_a)H(\mathbf{s}_a), \tag{44}$$

which then results in (5a).

For non-systematic codes, a formal proof involves information theoretic argument involving a periodic erasure channel. It is provided in Appendix II.

VII. Proof of Theorem 3

To prove Theorem 3, it suffices to establish the achievability at two points:

$$\bar{R}_a = \frac{B}{T_b+2B}, \tag{45a}$$

$$\bar{R}_b = \frac{T_b}{T_b+2B} \tag{45b}$$

and

$$\tilde{R}_a = \frac{2B}{T_a+B}, \tag{46a}$$

$$\tilde{R}_b = \frac{T_a-2B}{T_a+B} \tag{46b}$$

are achievable. The rest of the boundary can be achieved using "time-sharing" with the extreme points given by single-stream codes as in Remark 2.

The first corner point, $(\bar{R}_a, \bar{R}_b)$, in (45) can be achieved by reducing the value of the delay of stream a to $T_a^\star = T_b+B < T_a$ and use the low-delay regime construction in Section VI which achieves the rate pair,

$$\left(\frac{B}{T_a^\star+B}, \frac{T_a^\star-B}{T_a^\star+B}\right) = \left(\frac{B}{T_b+2B}, \frac{T_b}{T_b+2B}\right) = (\bar{R}_a, \bar{R}_b). \tag{47}$$

The second corner point, $(\tilde{R}_a, \tilde{R}_b)$, in (45) can be achieved by reducing the value of T_b to $T_b^\star = T_a-2B < T_b$ and use the large-delay regime construction in Section V which achieves the rate pair,

$$\left(\frac{T_a-T_b^\star}{T_a+B}, \frac{T_b^\star}{T_a+B}\right) = \left(\frac{2B}{T_a+B}, \frac{T_a-2B}{T_a+B}\right) = (\tilde{R}_a, \tilde{R}_b). \tag{48}$$

We remark that the upper bound is the same as in Theorem 1 for systematic codes; see Remark 3.

[1]Alternately we can use the recovery condition in (11)

Figure	R	k_b	k_a	R_b	R_a	B	B_{ms}	Regime
Fig. 4	5/7	3	2	3/7	2/7	4	2	Low-Delay
Fig. 5	11/16	6	5	6/16	5/16	5	3	Medium-Delay
Fig. 6	5/7	1	4	1/7	4/7	8	6	Medium-Delay
Fig. 7	3/4	1	2	1/4	2/4	8	4	Large-Delay

TABLE I: Code parameters used in the simulations.

VIII. Simulation Results

In this section, we compare the performance of the proposed codes to existing codes over statistical channel models such as Gilbert and Extended-Gilbert channels. A Gilbert channel is a two-state Markov model. In the "good state", each channel packet is received perfectly, whereas in the "bad state" each channel packet is lost with a probability of 1. We will use the symbols α and β to denote the transition probabilities from the good state to the bad state and vice versa. As long as the channel stays in the bad state the channel behaves as a burst-erasure channel. The length of each burst is a Geometric random variable of mean $\frac{1}{\beta}$ while the gap between successive bursts is also a geometric random variable with mean $\frac{1}{\alpha}$. An extended Gilbert-Channel is an extension of the Gilbert channel model with a total of $\mathcal{N}+1$ states, denoted by $\{0, 1, \ldots, \mathcal{N}\}$. State '0' is the good state and the remaining $\mathcal{N}$ states are bad states. We let the transition probability from the good state to the first bad state to be α whereas the transition probability from state i to state $(i+1)$, as well as the transition probability from state $\mathcal{N}$ to state 0 equals β. The burst length distribution in such a model is a hyper-geometric random variable as it is a sum of $\mathcal{N}$ i.i.d. geometric random variables. We will consider a setting where two source streams with different delays are multiplexed into a single channel packet stream. We denote the rates by R_a and R_b. Our baseline scheme is the MS code applied separately to the two streams in Section IV-B. The parameters are outlined in Table I.

In Fig. 4 and 5, we consider a Gilbert channel with $\alpha = 10^{-3}$ and $\beta \in [0.3, 0.7]$. We plot β on the x-axis and the residual packet loss rate on the y-axis. In Fig. 4, the rate for the two streams are $R_a = 2/7$ and $R_b = 3/7$ while the associated delays are $T_a = 10$ and $T_b = 6$. It can be shown that this choice of parameters corresponds to the low delay regime

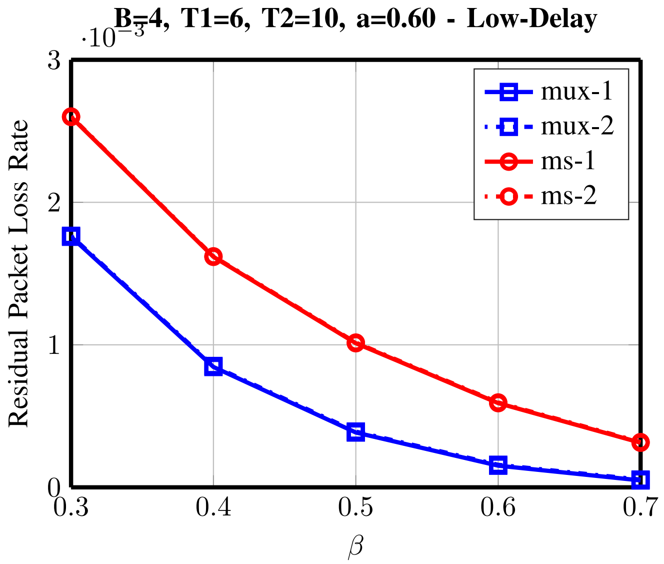


Fig. 4: Simulation Results over Gilbert Channel model in the low delay regime.

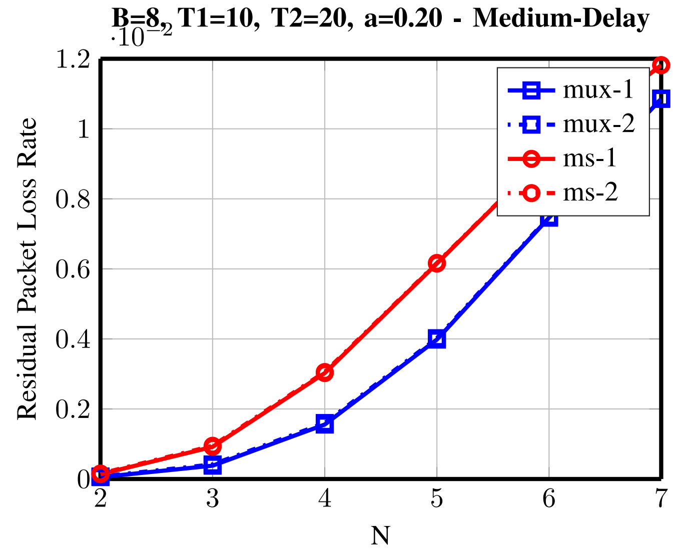


Fig. 6: Simulation Results over the Extended Gilbert Channel model in the medium delay regime.

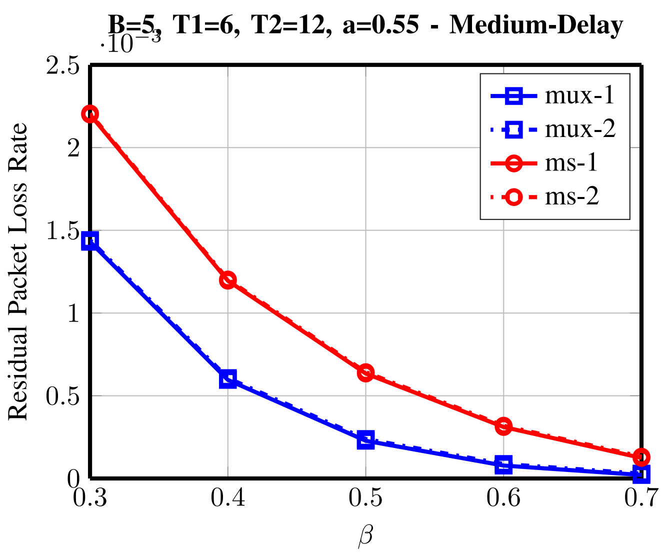


Fig. 5: Simulation Results over Gilbert Channel model in the medium delay regime.

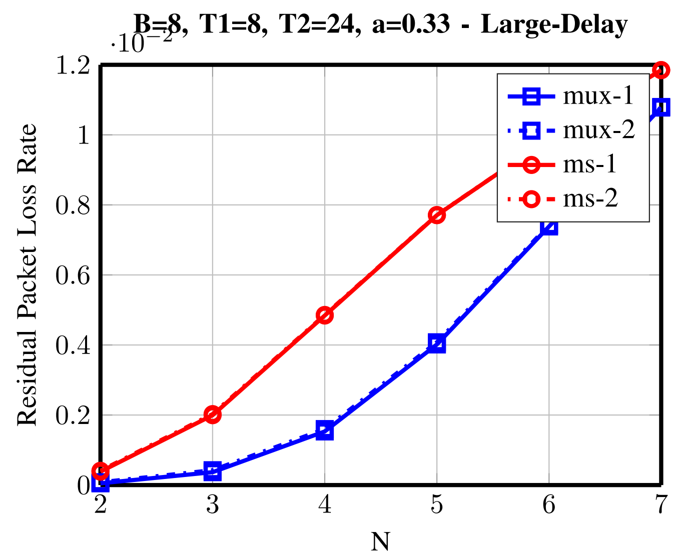


Fig. 7: Simulation Results over the Extended Gilbert Channel model in the large delay regime.

in Section VI and maximum recoverable burst length for our proposed $B = 4$. The MS code in Section IV-B achieves a $B = 2$ for these parameters. This explains the gap in the residual loss rates in Fig. 4. We note that both the streams exhibit the same residual loss rate for each of the two codes, as they are affected by the same set of long burst patterns.

In Fig. 5, the rates of the two streams are $R_b = 6/16$ and $R_a = 5/16$ and the delays are $T_b = 6$ and $T_a = 12$. For this choice of parameters the recoverable burst length achieved by the optimal code is $B = 5$ and uses the intermediate-delay construction in (45). The MS code achieves $B = 3$ at this rate and delay constraints and hence achieves a larger residual loss rate.

In Fig. 6 and 7, we consider an Extended-Gilbert channel with $\alpha = 10^{-3}$ and $\beta = 0.5$. We plot the number of states N on the x-axis and the residual packet loss rate on the y-axis. In Fig. 6, the rates for the two streams are $R_a = 4/7$ and $R_b = 1/7$ and the associated delays are $T_b = 10$ and $T_a = 20$. The optimal burst length is 8 compared to the the MS code, that achieves $B = 6$. In Fig. 7, the rates of the two streams are $R_a = 2/4$ and $R_b = 1/4$ and the associated delays are $T_b = 8$ and $T_a = 24$. The optimal burst length that can be achieved is $B = 8$ while the MS code only achieves $B = 4$.

IX. Conclusions

Error correction codes for streaming applications are different from traditional codes in many fundamental ways. In this paper we introduce the problem of multiplexing two source streams with different decoding delays into a single channel packet stream. We consider a burst erasure channel that introduces an erasure burst of maximum length B, and develop a new class of streaming codes that apply joint FEC across the two streams. The nature of our coding scheme depends on the relative magnitude of the delays in the two streams. We propose three operating regimes for the system and propose a coding scheme for each case. We also establish the optimality of our proposed codes for a certain range of system parameters. We note that simply encoding the source streams separately using previously proposed single-stream codes, and concatenating the resulting outputs is suboptimal in general.

As future work one can consider channels that introduce both burst and isolated losses. Our constructions can be naturally extended using the layered coding framework in [2], but the optimality remains to be verified. One can consider rank-metric extensions of these codes. Such extensions would be of interest if the underlying network applies a linear network code, while the streaming code is applied at the end nodes. In the single stream case such extensions have been proposed in [3]. Furthermore in our setup we considered the case when the source and transmission rates are matched, i.e., in each slot one packet from each source stream arrives and one channel packet is transmitted. One can also consider the mismatched settings where multiple channel packets are transmitted between successive source packet, along the lines of [2], [4], [11]. Finally one can also consider settings with more than two source streams, multiple parallel links between the source and destination, as well as providing unequal error protection for the two streams.

APPENDIX I
PROOF OF (29)

We use mathematical induction. For the base case we substitute we substitute $r = T_a$, we have

$$\begin{aligned}
&\sum_{i=0}^{T_a-1} H(\mathbf{x}[i]) \\
&\geq H\Big(\mathbf{s}_a\begin{bmatrix} -1 \\ 0 \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} T_a-B-1 \\ T_a-B \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} T_a-1 \\ T_a \end{bmatrix}\Big) \\
&\quad + H\Big(\mathbf{x}\begin{bmatrix} T_a-1 \\ 0 \end{bmatrix}\Big|\mathbf{s}_a\begin{bmatrix} -1 \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} T_a-B-1 \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} T_a-1 \\ T_a \end{bmatrix}\Big) \\
&= H\Big(\mathbf{x}\begin{bmatrix} T_a-1 \\ 0 \end{bmatrix}\Big)
\end{aligned} \tag{49}$$

which is true and establishes the base case. Before conducting the induction step, we first take $H(x[i])$ where $i \geq B$:

$$\begin{aligned}
&H(\mathbf{x}[i]) \geq H\Big(\mathbf{x}[i]\Big|\mathbf{x}\begin{bmatrix} i-B-1 \\ 0 \end{bmatrix}\Big) \\
&= H\Big(\mathbf{s}_b[i-B]\mathbf{x}[i]\Big|\mathbf{x}\begin{bmatrix} i-B-1 \\ 0 \end{bmatrix}\Big) \\
&\qquad - H\Big(\mathbf{s}_b[i-B]\Big|\mathbf{x}[i]\mathbf{x}\begin{bmatrix} i-B-1 \\ 0 \end{bmatrix}\Big) \\
&\overset{(a)}{=} H\Big(\mathbf{s}_b[i-B]\mathbf{x}[i]\Big|\mathbf{x}\begin{bmatrix} i-B-1 \\ 0 \end{bmatrix}\Big) \\
&\overset{(b)}{=} H(\mathbf{s}_b[i-B]) + H\Big(\mathbf{x}[i]\Big|\mathbf{s}_b[i-B]\mathbf{x}\begin{bmatrix} i-B-1 \\ 0 \end{bmatrix}\Big)
\end{aligned} \tag{50}$$

where (a) uses the fact that $\mathbf{s}_b[i-B]$ is recovered from $\mathbf{x}[i], \mathbf{x}\begin{bmatrix} i-B-1 \\ 0 \end{bmatrix}$, and (b) uses the fact that $\mathbf{s}_b[i-B]$ is independent of all source-packets and also all channel-packets with time index less than $i-B$.

Now let us assume that (29) is true for $r = q$ where $q \geq T_a$. When we combine this with (50), we have

$$\begin{aligned}
&\sum_{i=0}^{q} H(\mathbf{x}[i]) = \sum_{i=0}^{q-1} H(\mathbf{x}[i]) + H(\mathbf{x}[q]) \\
&\geq H\Big(\mathbf{s}_a\begin{bmatrix} q-T_a-1 \\ 0 \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-B-1 \\ T_a-B \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\quad + H\Big(\mathbf{x}\begin{bmatrix} q-1 \\ 0 \end{bmatrix}\Big|\mathbf{s}_a\begin{bmatrix} q-T_a-1 \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-B-1 \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\quad + H(\mathbf{s}_b[q-B]) + H\Big(\mathbf{x}[q]\Big|\mathbf{s}_b[q-B]\mathbf{x}\begin{bmatrix} q-B-1 \\ 0 \end{bmatrix}\Big) \\
&\geq H\Big(\mathbf{s}_a\begin{bmatrix} q-T_a-1 \\ 0 \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\quad + H\Big(\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}\Big|\mathbf{s}_a\begin{bmatrix} q-T_a-1 \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&= H\Big(\mathbf{s}_a\begin{bmatrix} q-T_a-1 \\ 0 \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\quad + H\Big(\mathbf{s}_a[q-T_a]\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}\Big|\mathbf{s}_a\begin{bmatrix} q-T_a-1 \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\quad - H\Big(\mathbf{s}_a[q-T_a]\Big|\mathbf{s}_a\begin{bmatrix} q-T_a-1 \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}\Big) \\
&\overset{(a)}{=} H\Big(\mathbf{s}_a\begin{bmatrix} q-T_a-1 \\ 0 \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\quad + H\Big(\mathbf{s}_a[q-T_a]\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}\Big|\mathbf{s}_a\begin{bmatrix} q-T_a-1 \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\overset{(b)}{=} H\Big(\mathbf{s}_a\begin{bmatrix} q-T_a \\ 0 \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\quad + H\Big(\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}\Big|\mathbf{s}_a\begin{bmatrix} q-T_a \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&= H\Big(\mathbf{s}_a\begin{bmatrix} q-T_a \\ 0 \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\quad + H\Big(\mathbf{s}_b[q]\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}\Big|\mathbf{s}_a\begin{bmatrix} q-T_a \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\quad - H\Big(\mathbf{s}_b[q]\Big|\mathbf{s}_a\begin{bmatrix} q-T_a \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}\Big) \\
&\overset{(c)}{=} H\Big(\mathbf{s}_a\begin{bmatrix} q-T_a \\ 0 \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\quad + H\Big(\mathbf{s}_b[q]\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}\Big|\mathbf{s}_a\begin{bmatrix} q-T_a \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-1 \\ T_a \end{bmatrix}\Big) \\
&\overset{(d)}{=} H\Big(\mathbf{s}_a\begin{bmatrix} q-T_a \\ 0 \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\Big) + H\Big(\mathbf{s}_b\begin{bmatrix} q \\ T_a \end{bmatrix}\Big) \\
&\quad + H\Big(\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}\Big|\mathbf{s}_a\begin{bmatrix} q-T_a \\ 0 \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q-B \\ T_a-B \end{bmatrix}\mathbf{s}_b\begin{bmatrix} q \\ T_a \end{bmatrix}\Big)
\end{aligned} \tag{51}$$

where step (a) uses the fact that $\mathbf{s}_a[q-T_a]$ is recovered from $\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}$, (b) uses the fact that $\mathbf{s}_a[q-T_a]$ is independent of all source and channel-packets before time $q-T_a$, (c) uses the fact that $\mathbf{s}_b[q]$ must be recovered from $\mathbf{x}\begin{bmatrix} q \\ 0 \end{bmatrix}$ when there is an erasure burst spanning $[q+1, q+B]$, and (d) uses the fact that source-packets are independent of each other. The result is the formula in (29) for $r = q+1$.

APPENDIX II
CONVERSE PROOF OF THEOREM 2

The argument used to prove (4b) can be used to prove (5b). We now prove the inequality in (5a). We start by defining a periodic erasure channel with a period of $T_b + B$ packets. The first B packets are erased whereas the remaining T_b packets are not. For any $j \geq 0$, the indices of the packets in the j-th

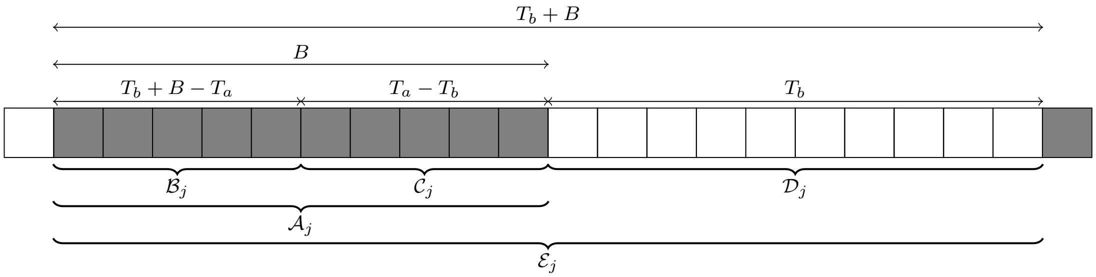


Fig. 8: The jth period of the periodic erasure channel used in proving (5a) in Theorem 2. Grey and white squares denote erased and unerased packets respectively. The sets of indices in (52) are also shown.

period are grouped as follows,

$$\mathcal{A}_j = [j(T_b+B), j(T_b+B)+B-1], \tag{52a}$$
$$\mathcal{B}_j = [j(T_b+B), j(T_b+B)+T_b+B-T_a-1], \tag{52b}$$
$$\mathcal{C}_j = [j(T_b+B)+T_b+B-T_a, j(T_b+B)+B-1], \tag{52c}$$
$$\mathcal{D}_j = [j(T_b+B)+B, (j+1)(T_b+B)-1], \tag{52d}$$
$$\mathcal{E}_j = [j(T_b+B), (j+1)(T_b+B)-1], \tag{52e}$$

and are shown in Fig. 8 at $j=0$. For the sake of simplicity, we use the notation $\mathcal{A}_{j_1}^{j_2}$ to denote,

$$\mathcal{A}_{j_1}^{j_2} = \begin{cases} \cup_{j=j_1}^{j_2} \mathcal{A}_j, & j_2 \geq j_1, \\ \emptyset, & otherwise, \end{cases} \tag{53}$$

and $\mathbf{s}(\mathcal{A}_j)$ to denote the set of source symbols with indices in $\mathcal{A}_j$.

Through a direct application of (1), one can write,

$$H\left(\mathbf{s}_b(\mathcal{A}_j)\middle|\mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_j)\right) = 0, \tag{54a}$$
$$H\left(\mathbf{s}_a(\mathcal{B}_j)\middle|\mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_j)\right) = 0, \tag{54b}$$
$$H\left(\mathbf{s}(\mathcal{D}_j)\middle|\mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{s}(\mathcal{A}_j), \mathbf{x}(\mathcal{D}_{j+1})\right) = 0, \tag{54c}$$

which are useful in the following inequalities

$$\begin{aligned}
&rBk_b + r(T_b+B-T_a)k_a + rT_b(k_a+k_b) \\
&= H\left(\mathbf{s}_b(\mathcal{A}_0^{r-1}), \mathbf{s}_a(\mathcal{B}_0^{r-1}), \mathbf{s}(\mathcal{D}_0^{r-1})\right) && \text{(55a)} \\
&= H\left(\mathbf{s}_b(\mathcal{A}_0^{r-1}), \mathbf{s}_a(\mathcal{B}_0^{r-1}), \mathbf{s}_a(\mathcal{C}_0^{r-1}), \mathbf{s}(\mathcal{D}_0^{r-1})\right) \\
&\quad - H\left(\mathbf{s}_a(\mathcal{C}_0^{r-1})\right) && \text{(55b)} \\
&\leq H\left(\mathbf{s}_b(\mathcal{A}_0^{r-1}), \mathbf{s}_a(\mathcal{B}_0^{r-1}), \mathbf{s}_a(\mathcal{C}_0^{r-1}), \mathbf{s}(\mathcal{D}_0^{r-1}), \mathbf{x}(\mathcal{D}_0^{r})\right) \\
&\quad - H\left(\mathbf{s}_a(\mathcal{C}_0^{r-1})\right) \\
&= H\left(\mathbf{s}(\mathcal{E}_0^{r-1}), \mathbf{x}(\mathcal{D}_0^{r})\right) - H\left(\mathbf{s}_a(\mathcal{C}_0^{r-1})\right) \\
&= H\left(\mathbf{x}(\mathcal{D}_0^{r})\right) + \sum_{j=0}^{r-1}\Big(H\left(\mathbf{s}_b(\mathcal{A}_j)\middle|\mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_0^{r})\right) \\
&\quad +H\left(\mathbf{s}_a(\mathcal{B}_j)\middle|\mathbf{s}_b(\mathcal{A}_j), \mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_0^{r})\right) \\
&\quad +H\left(\mathbf{s}_a(\mathcal{C}_j)\middle|\mathbf{s}_b(\mathcal{A}_j), \mathbf{s}_a(\mathcal{B}_j), \mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_0^{r})\right) \\
&\quad -H\left(\mathbf{s}_a(\mathcal{C}_j)\middle|\mathbf{s}_a(\mathcal{C}_0^{j-1})\right) \\
&\quad +H\left(\mathbf{s}(\mathcal{D}_j)\middle|\mathbf{s}_b(\mathcal{A}_j), \mathbf{s}_a(\mathcal{B}_j), \mathbf{s}_a(\mathcal{C}_j), \mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_0^{r})\right)\Big) \\
&\leq H\left(\mathbf{x}(\mathcal{D}_0^{r})\right) + \sum_{j=0}^{r-1}\Big(H\left(\mathbf{s}_b(\mathcal{A}_j)\middle|\mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_0^{r})\right) \\
&\quad +H\left(\mathbf{s}_a(\mathcal{B}_j)\middle|\mathbf{s}_b(\mathcal{A}_j), \mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_0^{r})\right) \\
&\quad +H\left(\mathbf{s}(\mathcal{D}_j)\middle|\mathbf{s}_b(\mathcal{A}_j), \mathbf{s}_a(\mathcal{B}_j), \mathbf{s}_a(\mathcal{C}_j), \mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_0^{r})\right)\Big) && \text{(55c)} \\
&\leq H\left(\mathbf{x}(\mathcal{D}_0^{r})\right) + \sum_{j=0}^{r-1}\Big(H\left(\mathbf{s}_b(\mathcal{A}_j)\middle|\mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_0^{r})\right) \\
&\quad +H\left(\mathbf{s}_a(\mathcal{B}_j)\middle|\mathbf{s}_b(\mathcal{A}_j), \mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_0^{r})\right) \\
&\quad +H\left(\mathbf{s}(\mathcal{D}_j)\middle|\mathbf{s}_b(\mathcal{A}_j), \mathbf{s}_a(\mathcal{B}_j), \mathbf{s}_a(\mathcal{C}_j), \mathbf{s}(\mathcal{E}_0^{j-1}), \mathbf{x}(\mathcal{D}_0^{r})\right)\Big) \\
&= H\left(\mathbf{x}(\mathcal{D}_0^{r})\right) && \text{(55d)} \\
&\leq (r+1)T_b n, && \text{(55e)}
\end{aligned}$$

where (55a) and (55b) follows from the independence of source symbols, (55c) uses the fact that conditioning reduces the entropy and also

$$H\left(\mathbf{s}_a(\mathcal{C}_j)\middle|\mathbf{s}_a(\mathcal{C}_0^{j-1})\right) = H\left(\mathbf{s}_a(\mathcal{C}_j)\right)$$

since the source packets are i.i.d., and (55d) uses (54).

By substituting $R_b = k_b/n$ and $R_a = k_a/n$ in (55e), we get

$$\left(\frac{B+T_b}{T_b}\right)R_b + \left(\frac{2T_b+B-T_a}{T_b}\right)R_a \leq \frac{r+1}{r} \xrightarrow{r\to\infty} 1 \tag{56}$$

and (5a) follows.

References

[1] E. Martinian and C. W. Sundberg, “Burst Erasure Correction Codes with Low Decoding Delay,” *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2494–2502, 2004.

[2] A. Badr, P. Patil, A. Khisti, W. Tan, and J. Apostolopoulos, “Layered Constructions for Low-Delay Streaming Codes,” *IEEE Transactions on Information Theory*, 2017.

[3] R. Mahmood, A. Badr, and A. Khisti, “Convolutional codes with maximum column sum rank for network streaming,” *IEEE Transactions on Information Theory*, vol. 62, no. 6, pp. 3039–3052, 2016.

[4] P. Patil, A. Badr, and A. Khisti, “Delay-optimal streaming codes under source-channel rate mismatch,” in *Proc. Asilomar Conference on Signals, Systems & Computers*, 2013.

[5] A. Badr, A. Khisti, and E. Martinian, “Diversity Embedded Streaming erasure Codes (DE-SCo): Constructions and Optimality,” *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 29, no. 5, pp. 1042–1054, May 2011.

[6] A. Badr, A. Khisti, W.-t. Tan, and J. Apostolopoulos, "Streaming codes with partial recovery over channels with burst and isolated erasures," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 3, pp. 501–516, 2015.

[7] A. Badr, "Error-Correcting Codes for Low-Delay Streaming Communications," Ph.D. dissertation, University of Toronto, 2014.

[8] A. Badr, D. Lui, and A. Khisti, "Streaming-Codes for Multicast over Burst Erasure Channels," *IEEE Transactions on Information Theory*, 2016.

[9] A. Badr, A. Khisti, W. Tan, and J. Apostolopoulos, "Streaming Codes for Channels with Burst and Isolated Erasures," in *Proc. International Conference on Computer Communications (INFOCOM)*, 2013.

[10] O. Tekin, T. Ho, H. Yao, and S. Jaggi, "On erasure correction coding for streaming," in *Information Theory and Applications Workshop (ITA)*, 2012, pp. 221–226.

[11] D. Leong and T. Ho, "Erasure coding for real-time streaming," in *Proc. International Symposium on Information Theory (ISIT)*, 2012.

[12] Z. Li, A. Khisti, and B. Girod, "Correcting erasure bursts with minimum decoding delay," in *Proc. Asilomar Conference on Signals, Systems & Computers*, 2011.

[13] D. Lui, "Coding theorems for delay sensitive communication over burst-erasure channels," Master's thesis, University of Toronto, Toronto, ON, August 2011.

[14] Y. Li and E. Soljanin, "Rateless codes for single-server streaming to diverse users," *CoRR*, vol. abs/0912.5055, 2009.

[15] H. Yao, Y. Kochman, and G. W. Wornell, "A multi-burst transmission strategy for streaming over blockage channels with long feedback delay," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 29, no. 10, pp. 2033–2043, 2011.

[16] G. Joshi, Y. Kochman, and G. Wornell, "On playback delay in streaming communications," in *Proc. International Symposium on Information Theory (ISIT)*, 2012.

[17] S. Mehrotra, J. Li, and Y. Huang, "Optimizing fec transmission strategy for minimizing delay in lossless sequential streaming," *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 1066–1076, 2011.

[18] A. Talari and N. Rahnavard, "On the intermediate symbol recovery rate of rateless codes," *IEEE Transactions on Communications*, vol. 60, no. 5, pp. 1237–1242, 2012.

[19] B. N. Vellambi, N. Torabkhani, and F. Fekri, "Throughput and latency in finite-buffer line networks," *IEEE Transactions on Information Theory*, vol. 57, no. 6, pp. 3622–3643, 2011.

[20] N. Rahnavard, B. N. Vellambi, and F. Fekri, "Rateless codes with unequal error protection property," *IEEE Transactions on Information Theory*, vol. 53, no. 4, pp. 1521–1532, 2007.

[21] R. Subramanian and F. Fekri, "Analysis of latency and related tradeoffs in distribute sensor networks," in *IEEE International Conference on Sensing, Communication, and Networking (SECON)*, 2007, pp. 391–400.

[22] G. Carlucci, L. De Cicco, and S. Mascolo, "HTTP over UDP: an Experimental Investigation of QUIC," in *Proceedings of the 30th Annual ACM Symposium on Applied Computing*. ACM, 2015, pp. 609–614.

[23] E. Martinian and M. Trott, "Delay-optimal burst erasure code construction," in *Proc. International Symposium on Information Theory (ISIT)*, Nice, France, July 2007.

[24] Y. Li, P. Vingelmann, M. Pedersen, and E. Soljanin, "Round-robin streaming with generations," *CoRR*, vol. abs/1206.3014, 2012.

[25] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4413–4430, 2006.

[26] H. Gluesing-Luerssen, J. Rosenthal, and R. Smarandache, "Strongly-MDS Convolutional Codes," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 584–598, 2006.