Streaming Erasure Codes over Multi-Access Relay Networks

Gustavo Kasper Facenda[†], Elad Domanovitz[†], Ashish Khisti[†], Wai-Tian Tan[*] and John Apostolopoulos[*]
[†] University of Toronto, Toronto, ON M5S 3G4, Canada
E-mails: {gustavo.kasperfacenda@mail.utoronto.ca, elad.domanovitz@utoronto.ca, akhisti@ece.utoronto.ca}
[*] Enterprise Networking Innovation Labs, Cisco Systems, San Jose, CA 95134, USA

***Abstract*—Applications where multiple users communicate with a common server and desire low latency are common and increasing. This paper studies a network with two source nodes, one relay node and a destination node, where each source nodes wishes to transmit a sequence of messages, through the relay, to the destination, who is required to decode the messages with a strict delay constraint T. The network with a single source node has been studied in [1]. We start by introducing two important tools: the delay spectrum, which generalizes delay-constrained point-to-point transmission, and concatenation, which, similar to time sharing, allows combinations of different codes in order to achieve a desired regime of operation. Using these tools, we are able to generalize the two schemes previously presented in [1], and propose a novel scheme which allows us to achieve optimal rates under a set of well-defined conditions. Such novel scheme is further improved in order to achieve higher rates in the scenarios where the conditions for optimality are not met.**

I. Introduction

A number of emerging applications including online real-time gaming, real-time video streaming (video conference with multiple users) and healthcare (under the name tactile internet) require efficient low-latency communication. In these applications, data packets are generated at the source in a sequential fashion and must be transmitted to the destination under strict latency constraints. When packets are lost over the network, significant amount of error propagation can occur and suitable methods for error correction are necessary.

There are two main approaches for error correction due to packet losses in communication networks: Automatic repeat request (ARQ) and Forward error correction (FEC). ARQ is inherently inferior when considering low latency constraints, especially for long distance communication, and, for that reason, FEC schemes are considered more appropriate candidates. The literature has studied codes with strict decoding-delay constraints—called streaming codes—in order to establish fundamental limits of reliable low-latency communication under a variety of packet-loss models. Previous works have studied particular, useful cases. In [2], the authors studied a point-to-point (i.e., two nodes—source and destination) network under a maximal burst erasure pattern. In [3], the authors have studied, separately, burst erasures and arbitrary erasures. In [4], the authors have extended the erasure pattern, allowing for both burst erasures and arbitrary erasures. In particular, it was shown that random linear codes [5] are optimal if we are concerned only with correcting arbitrary erasures. Other works that have further studied various aspects of low-latency streaming codes include [6]–[14].

While most of the prior work on streaming codes has focused on a point-to-point communication link, a network topology that is of practical interest involves a relay node between source and destination, that is, a three-node network. This topology is motivated by numerous applications in which a gateway server, able to decode and encode data, connects two end nodes. Motivated by such considerations, streaming codes for such a setting were first introduced in [1] and further extended to a multi-hop network in [15].

However, a significant part of the mentioned applications, such as real-time gaming and video conferences, involve communications between multiple users and a common server. Motivated by such applications, in this paper we extend the relayed topology of [1] for a multiple access relay channel (MARC). We focus on the case with two source nodes in this paper, but it should be noted that the proposed schemes and the converse can be directly extended to multiple source nodes, although the expressions become hard to evaluate. We focus our analysis on the scenario where the link connecting the relay and the destination represents the bottleneck, and we study the achievable rate region under such topology. This analysis is extended to any scenario in the full paper [16], where it can be seen that the scenario studied in this paper is the most challenging one. The proofs are omitted in this paper due to space limitations, but are presented in [16].

A. Main contributions

In this paper, considering a network with few assumptions on the parameters, we

- Develop a generalization of the concept of the delay spectrum of point-to-point codes and present an achievable delay spectrum for such codes.
- Present an upper bound on the achievable rate region of the two-user MARC under arbitrary erasures with a strict delay constraint T.
- Present a time-sharing-like tool for symbol-wise decode-and-forward (SWDF) streaming codes for the MARC.
- Propose two schemes using the tools we have developed: Concatenated SWDF (CSWDF) and Fixed Bottleneck SWDF (FB-SWDF). Further, we propose to use the time-sharing tool in order to obtain a result better than the maximum between both.

2021 IEEE International Symposium on Information Theory (ISIT) | 978-1-5386-8209-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISIT45174.2021.9518192

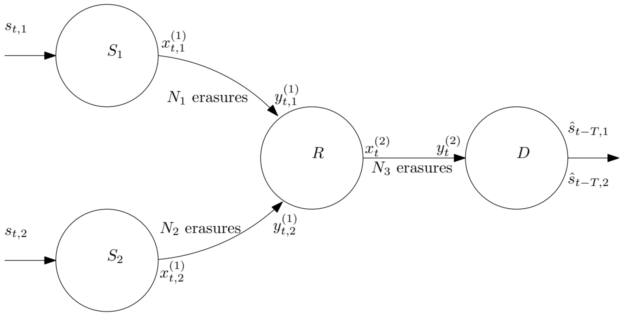

Fig. 1: Multiple Access Relay Channel

- Derive the conditions for the upper bound to be achievable using our proposed schemes.
- Compare the proposed schemes to a naive solution, or baseline scheme, denoted as Concatenated Message-Wise Decode-and-Forward (CMWDF), and demonstrate that our proposed scheme significantly outperforms the naive solution in terms of achievability.

In [16], we further extend the analysis and proposed schemes to remove the aforementioned assumptions, and we also present an improvement, denoted as Optimized Bottleneck SWDF (OB-SWDF), which improves upon FB-SWDF.

II. Preliminaries

In this paper, we consider a network with two sources, one relay and one destination. Each source i wishes to transmit a sequence of messages $\{s_{t,i}\}_{t=0}^{\infty}$ to the destination through a common relay. We assume that there is no direct link between sources and destination, which is an important distinction to the classic MARC (see, e.g., [17]). We assume that the link between the ith source and the relay introduces at most N_i erasures, and the link between relay and destination introduces at most N_3 erasures. The destination wishes to decode both source packets with a common delay T. This setting is illustrated in Fig. 1. Finally, we assume no processing or transmission delay (all operations happen at the same time instant) and the time-slots are synchronized across all nodes.

In the following, we present the formal definitions for the problem. For simplicity, we define $\mathbb{F}_e^n = \mathbb{F}^n \cup \{*\}$. The following definitions are standard and a straight-forward generalization of [1].

Definition 1. *An $(n_1, n_2, n_3, k_1, k_2, T)_{\mathbb{F}}$-streaming code consists of the following:*

- *Two sequences of source messages $\{s_{t,1}\}_{t=0}^{t=\infty}$ and $\{s_{t,2}\}_{t=0}^{t=\infty}$, where $s_{t,i} \in \mathbb{F}^{k_i}$.*
- *Two encoding functions $f_{t,i} : \underbrace{\mathbb{F}^{k_i} \times \cdots \times \mathbb{F}^{k_i}}_{t+1 \text{ times}} \to \mathbb{F}^{n_i}$, $i \in \{1,2\}$ each used by its respective source i at time t to generate $x_{t,i}^{(1)} = f_{t,i}(s_{0,i}, s_{1,i}, \ldots, s_{t,i})$.*
- *A relaying function $g_t : \underbrace{\mathbb{F}_e^{n_1} \times \cdots \mathbb{F}_e^{n_1}}_{t+1 \text{ times}} \times \underbrace{\mathbb{F}_e^{n_2} \times \cdots \mathbb{F}_e^{n_2}}_{t+1 \text{ times}} \to \mathbb{F}^{n_3}$ used by the relay at time t to generate $x_t^{(2)} = g_t(\{y_{j,1}^{(1)}\}_{j=0}^t, \{y_{j,2}^{(1)}\}_{j=0}^t)$.*
- *Two decoding functions ($i \in \{1,2\}$): $\varphi_{t+T,i} = \underbrace{\mathbb{F}_e^{n_3} \times \cdots \times \mathbb{F}_e^{n_3}}_{t+T+1 \text{ times}} \to \mathbb{F}^{k_i}$ used by the destination at time $t+T$ to generate two estimates $\hat{s}_{t,i} = \varphi_{t+T,i}(y_0^{(2)}, y_1^{(2)}, \ldots, y_{t+T}^{(2)})$.*

Definition 2. *An erasure sequence is a binary sequence denoted by $e^{\infty} \triangleq \{e_t\}_{t=0}^{\infty}$, where $e_t = \mathbf{1}\{$an erasure occurs at time $t\}$. An N-erasure sequence is an erasure sequence e^{∞} that satisfies $\sum_{t=0}^{\infty} e_t^{\infty} = N$. The set of N-erasure sequences is denoted by Ω_N.*

Definition 3. *The mapping $h_n : \mathbb{F}^n \times \{0,1\} \to \mathbb{F}_e^n$ of an erasure channel is defined as $h_n(x,e) = \begin{cases} x, & \text{if } e = 0 \\ *, & \text{if } e = 1 \end{cases}$.*

For any erasure sequence e^{∞} and any $(n_1, n_2, n_3, k_1, k_2, T)_{\mathbb{F}}$-streaming code, the following input-output relation holds for each $t \in \mathbb{Z}_+$: $y_{t,1}^{(1)} = h_{n_1}(x_{t,1}^{(1)}, e_{t,1}^{(1)})$ and $y_{t,2}^{(1)} = h_{n_2}(x_{t,2}^{(1)}, e_{t,2}^{(1)})$, where $e_{t,i}^{(1)} \in \Omega_{N_i}$, $i \in \{1,2\}$. Similarly, the following input-output relation holds for for each $t \in \mathbb{Z}_+$: $y_t^{(2)} = h_{n_3}(x_t^{(2)}, e_t^{(2)})$, where $e_t^{(2)} \in \Omega_{N_3}$.

Definition 4. *An $(n_1, n_2, n_3, k_1, k_2, T)_{\mathbb{F}}$-streaming code is said to be (N_1, N_2, N_3)-achievable if, for any $e_{t,i}^{(1)}$ and $e_t^{(2)}$, for all $t \in \mathbb{Z}_+$ and all $s_{t,i} \in \mathbb{F}^{k_i}$, we have $\hat{s}_{t,i} = s_{t,i}$.*

Definition 5. *The pair of rates of an $(n_1, n_2, n_3, k_1, k_2, T)_{\mathbb{F}}$-streaming code is $(R_1, R_2) = \left(\frac{k_1}{n}, \frac{k_2}{n}\right)$, where $n = \max(n_1, n_2, n_3)$.*

Definition 6. *The capacity rate region of an (N_1, N_2, N_3)-MARC network under delay constraint T is defined as the set of all rate pairs (R_1, R_2) such that there exists an (N_1, N_2, N_3)-achievable $(n_1, n_2, n_3, k_1, k_2, T)_{\mathbb{F}}$-streaming code, where (R_1, R_2) are defined as above.*

A. Upper bound

In this section, we use the results in [1] to present a simple upper bound on the achievable rate region. We denote by $C(T,N) = \frac{T+1-N}{T+1}$ the capacity of a single-link point-to-point channel [4]. For the remaining of the paper, we always refer to single-link codes. Since each user is transmitting its own message without cooperation, $R_1 \leq C(T-N_3, N_1)$ and $R_2 \leq C(T-N_3, N_2)$ are direct extensions from [1]. Furthermore, we can optimistically consider N_2 erasures in both links in the first hop, and obtain the following upper bound on the sumrate: $R_1 + R_2 \leq C(T-N_2, N_3)$.

In this paper, we make the following assumptions, which guarantee the desired regime of operation:

- $N_1 \overset{(a)}{\geq} N_2 \geq N_3$. (a) is without loss of generality.
- $T \geq \frac{1}{2}\left(\sqrt{N_1^2 - 4N_3(N_2 - N_3)} + N_1 + 2N_2 - 2\right)$.

In other words, these assumptions are required in order to all the bounds described in this section to be active at some point in the rate region. In [16], we analyze the different regimes of operation that may occur from the lack of such assumptions.

III. Motivating Example

Before we introduce the general schemes and results, let us consider an example which compares the schemes that are going to be presented in this paper. Let us consider a network with $N_1 = 3$, $N_2 = 2$, $N_3 = 1$ and $T = 6$. Further, let us assume we wish to allow the first user (i.e., source node $i = 1$) to transmit at its maximal rate $R_1 = C(T - N_3, N_1) = 0.5$. Such rate is certainly achievable from [1], if the second user does not transmit. Then, we attempt to answer the following question: what is the best rate the second user can achieve without interfering with the first user? In this section, we will see the answer to that question using each one of the schemes, which will be described in Section V.

First, what is the fundamental limit? By simply applying the upper bound, we get $R_2 \leq \frac{4}{5} - \frac{3}{6} = 0.3$. Now, let us see what we can, in fact, achieve. Under CMWDF, the first user would transmit with $k = 3$, $n_1 = 6$, and it would require $n_3 = 6$ from the second hop.It should be easy to see that, therefore, there is no remaining "gap" for the second user, and, indeed, using CMWDF, if we have $R_1 = 0.5$ in this network, the second user is unable to transmit. These parameters come from finding the best T' for $R_1 = \min\left(C(T', N_1), C(T - T', N_3)\right)$ [1].

On the other hand, under CSWDF, the first user would use $k' = 3$, $n_1' = 6$, but the relay can transmit using only $n_3' = 4$. These "base" codes are the codes that achieve $C(T - N_3, N_i)$ [1]. Therefore, there is some gap, and we can use this gap to transmit more symbols from the second user. In fact, by employing 5 concatenations of the mentioned code, we would have $k_1 = 15$, $n_1 = 30$, $n_3 = 20$. Then, we can include two concatenations of the single-user streaming code of the second user, which has parameters $k'' = 4$, $n_2'' = 6$, $n_3'' = 5$, thus, forming a code with parameters $k_2 = 8$, $n_2 = 12$, $n_3 = 10$. Finally, by concatenating both codes in the relay, we have $n_3 = 30$, which is clearly larger than both n_1 and n_2, thus, we define it as n. Finally, we have $R_1 = 15/30 = 0.5$ and $R_2 = 8/30 = 0.2667$. While this is significantly better than CMWDF, there is still some gap to the upper bound.

Finally, let us consider a slightly more complex streaming code, which is the result of our FB-SWDF scheme: for the first user, we will use 5 concatenations of the single-user capacity-achieving code, thus we get $k_1 = 15$, $n_1 = 30$. Among these 15 symbols, the relay can recover each 5 with delays 3, 4 and 5. Now, for the second user, let us consider the following strategy: $k_2 = 9$, $n_2 = 27$, which is obtained with 9 concatenations of a $(3, 1)$ code. The relay can recover all the 9 symbols transmitted by this source-node with delay 2. Finally, the relay can employ a code with $k = 24$ and $n_3 = 30$, which can transmit each 6 symbols with delays from 1 to 4, and can be obtained with 6 concatenations of $(5, 4)$ codes. Then, we match the delays of the symbols according to Table I. Finally, note that we have $R_1 = 15/30 = 0.5$ and $R_2 = 9/30 = 0.3$, that is, our scheme is able to achieve the upper bound. Note that, in this case, all codes used are simple diagonal interleaving MDS codes [4], however, it required us to "merge" the streams transmitted by each user in the relay, which previous schemes are unable to do, and requires an in-depth analysis of the delay spectrum of point-to-point codes, developed in this paper. Such merging can be seen in the table, where some symbols transmitted by user 2 are retransmitted in the remaining one slot along the symbols from user 1.

TABLE I: Number of symbols transmitted with each delay in each hop. Blue symbols are transmitted by the first user, while green symbols are transmitted by the second user.

$\mathbf{T}$ in the first hop \ $\mathbf{T}$ in the second hop	1	2	3	4
2	1	1	1	6
3			5	
4		5		
5	5			

IV. Symbol-wise Decode and Forward and Delay Spectrum

In order to present our coding scheme, first let us define the notion of delay spectrum for a point-to-point code. This notion exists and is mentioned in [1], however, in that paper, the authors define the delay spectrum through the delay profile of a streaming code. In this paper, we define it for any point-to-point code. Further, we make an in-depth analysis of said delay spectrum, which has not been made before. Similar analysis has been done in works such as [18], where a source wishes to transmit two streams with different delays to a destination, however, we generalize it for any number of different streams and delays, focusing on the arbitrary erasure channel.

Definition 7. *An $(n, k, \mathbf{T})_{\mathbb{F}}$ point-to-point code, where $\mathbf{T} = [T[1], \ldots, T[k]]$ is the delay spectrum of the code, consists of:*

1) *A streaming message $\{s_t\}_{t=0}^{\infty}$ and an encoder [4].*
2) *A list of k decoding functions $\varphi_{t+T[j]} = \underbrace{\mathbb{F}_e^n \times \cdots \mathbb{F}_e^n}_{t+T[j]+1 \text{ times}} \to \mathbb{F}^k$ used by the receiver at time $t+T[j]$ to generate $\hat{s}_t[j]$, that is, an estimate of the jth element of s_t.*

Definition 8. *An $(n, k, \mathbf{T})_{\mathbb{F}}$ point-to-point code is said to achieve $\mathbf{T}$ under N erasures if, for any $e_t' \in \Omega_N$, $\varphi_{t+T[j]}(h_n(x_0, e_0'), \ldots, h_n(x_{t+T[j]}, e_{t+T[j]}')) = s_t[j]$.*

For the relaying strategy, let us now introduce the concept of SWDF. In this strategy, the relaying function employed by the code first decodes the source packets transmitted by the source, and then encodes them again. This is an extension of the SWDF defined in [1] for the three-node network. However, the addition of a second source node adds some nuances to the strategy, as the messages relayed by the relay now must be multiplexed in some way. Below, we broadly define this strategy. A more complete definition can be found in [16].

Definition 9. *Assume the source nodes transmit their source messages $\{s_{t,i}\}_t^{\infty}$ to the relay using an $(n_i, k_i, \mathbf{T}_i^{(1)})_{\mathbb{F}}$ point-to-point code. Then, a relay is said to employ a SWDF if it also uses a point-to-point code, and each symbol used as a message symbol by the relay is the relay's delayed estimate of a source symbol from any of the two source nodes.*

For the remaining of the paper, we use $\mathbf{T}_i^{(1)}$ to denote the delay spectrum of the code used by the source node i and $\mathbf{T}^{(2)}$ to denote the delay spectrum of the code employed by the relay. It is easy to see that, using the SWDF strategy, the overall delay of a symbol with delay $T_s^{(1)}$ in the first hop and $T_s^{(2)}$ in the second hop is given by $T_s = T_s^{(1)} + T_s^{(2)}$.

Furthermore, we define a concatenation of point-to-point codes. Again, a more complete definition is given in [16].

Definition 10. *A concatenation of an $(n', k', \mathbf{T}')_{\mathbb{F}}$ point-to-point code with an $(n'', k'', \mathbf{T}'')_{\mathbb{F}}$ point-to-point code is an $(n' + n'', k' + k'', [\mathbf{T}', \mathbf{T}''])$ point-to-point code where the encoding and decoding functions are concatenations of the functions of the original codes.*

Furthermore, it should be clear that, if such codes achieve the delay spectra $\mathbf{T}'$ and $\mathbf{T}''$ under N erasures, the concatenated code achieves the delay spectrum $[\mathbf{T}', \mathbf{T}'']$ as well. This is formally stated and proven in [16].

Another useful operation that can be made is simply permuting the source symbols, and it should be clear that permuting the source symbols results in permuting the delay spectrum. Again, this is formally stated and proven in [16]. Then, since any permutation of an achievable delay spectrum is also achievable, we may describe the delay spectrum of a code by stating how many symbols are transmitted with some delay.

Definition 11. *Consider a delay spectrum $\mathbf{T} = [T[1], T[2], \dots, T[k]]$. An equally-delayed symbols grouping description of such delay spectrum is given by a list of tuples $\mathbf{G} = [(T^{(g)}[1], k^{(g)}[1]), \dots, (T^{(g)}[\ell^{(g)}], k^{(g)}[\ell^{(g)}])]$, where $\ell^{(g)}$ is the length of the list. For simplicity, we assume $T^{(g)}[1] \geq T^{(g)}[2] \geq \cdots \geq T^{(g)}[\ell^{(g)}]$. Furthermore, we define $\mathbf{T}^{(g)} = [T^{(g)}[1], \dots, T^{(g)}[\ell^{(g)}]]$ and $\mathbf{k}^{(g)} = [k^{(g)}[1], \dots, k^{(g)}[\ell^{(g)}]]$, where $\sum_{i=1}^{\ell^{(g)}} k^{(g)}[i] = k$.*

A. Point-to-point results for delay spectrum

In this Section, we present an achievability result for point-to-point codes in terms of delay spectrum.

Lemma 1 (Achievability)**.** *Let*

$$\mathbf{T}^{(g)} = \Big[T^{(g)}[1], T^{(g)}[2], \dots, \\ T^{(g)}[\ell^{(g)}]\Big] = [T^{(g)}[1], T^{(g)}[1] - 1, \dots, N+1, N\Big]$$

and $\frac{n-k}{N}$ be an integer. Then, there exists an $(n, k, \mathbf{T})_{\mathbb{F}}$ point-to-point code that can transmit $k^{(g)}[1] = n - \frac{T^{(g)}[1]}{N}(n-k)$ symbols with delay $T^{(g)}[1]$ and $k^{(g)}[j] = \frac{n-k}{N} \forall j \geq 2$.

The code that achieves such delay spectrum is a concatenation of two diagonal interleaving MDS codes [4]. The details of the code construction can be found in the proof in [16].

Lemma 1 can be used to derive the following corollary:

Corollary 1. *Assume that $\mathbf{k}^{(g)} \leq \mathbf{k}^{con}$ is a constraint. If*

$$k \leq n - n \cdot N \cdot \frac{\left(1 - \sum_{\ell=1}^{j-1} \frac{k^{con}[\ell]}{n}\right)}{T^{(g)}[j] + 1} \forall j \in \{1, 2, \dots, \ell^{(g)}\} \tag{1}$$

there exists an $(n, k, \mathbf{T})_{\mathbb{F}}$ point-to-point code that achieves the desired delay spectrum $\mathbf{T}^{(g)}$ under N erasures.

V. Achievable rate region

In this section, we present lower bounds on the capacity rate region. We first describe FB-SWDF, in which the relay-destination link (the bottleneck) attempts to transmit at its maximal rate. We then present a time-sharing-like tool, which defines that from any two streaming codes, another streaming code can be derived with achievable rates which are at least a convex combination of both rates. Further, this tool can be used to extend schemes used over a three-node network.

We note that the one such extension denoted by CSWDF is able to partially outperform FB-SWDF (which is due to the attempt of FB-SWDF to transmit at maximal rate in the relay-destination link). Therefore, the achievable region we characterize is the outcome of applying "time-sharing" between CSWDF and FB-SWDF.

A. FB-SWDF

For this scheme, we use the point-to-point results derived earlier and analyze the achievable rate region using symbol-wise decode and forward. The following Theorem describes the rate region achieved by employing SWDF in conjunction with the results described in Section IV. Specifically, this scheme attempts to use a point-to-point code with rate $R_{bn} = C(T - N_2, N_3)$ in the relay-destination link. Then, in order to achieve R_1, it starts by employing a code with rate $R_1 = C(T - N_3, N_1)$, and then erasing information symbols from this code. In all codes, we use $n = (T + 1 - N_3)(T + 1 - N - 2)c$, where c is an auxiliary constant. It can be shown through Lemma 1 that, by employing such codes, there are $\frac{R_1 \cdot n}{T+1-N_1-N_3}$ symbols with delays $\{N_1, N_1 + 1, \dots, T - N_3\}$ being transmitted through the link with N_1 erasures, while the relay can transmit up to $\frac{R_{bn} \cdot n}{T+1-N_2-N_3}$ symbols with delays $\{N_3, N_3 + 1, \dots, T - N_2\}$. Then, we employ Corollary 1 in order to find the maximum achievable R_2, using, as constraint, the number of symbols that can still be transmitted through the bottleneck. The role of the auxiliary constant c is to guarantee that the number of symbols in each timeslot is integer.

Theorem 1. *For any R_1, the following rate is achievable*

$$R_2 = \begin{array}{c}\min\big(C(T - N_3, N_2), \\ \qquad C(T - N_2, N_3) - C(T - N_3, N_1), R_2'\big)\end{array} \tag{2}$$

$$R_2' = \frac{N_2}{N_1} \left[\frac{N_1}{N_2} - 1 - R_1 + \frac{T + 1 - N_1 - N_3}{T + 1 - N_2} \right]. \tag{3}$$

In particular, the Theorem leads to the following Corollary, which presents a sufficient condition for the sumrate capacity to be achieved in at least one point, and the two corner points in which the capacity sumrate is achieved.

Corollary 2. *If $(T + 1 - N_3)(T + 1 - N_2 - N_1) \geq (T + 1 - N_3 - N_1)(N_2 - N_3)$, then the sumrate capacity $R_1 + R_2 = C(T - N_2, N_3)$ is achieved at at least one point of the rate region, which is $(R_1, R_2) = (C(T - N_3, N_1), C(T - N_2, N_3) - C(T - N_3, N_1))$.*

Further, the rate pair $(R_1, R_2) = \left(\frac{N_2 - N_3}{T+1-N_2}, \frac{T+1-2N_2}{T+1-N_2}\right)$ *is also achievable.*

B. Time-sharing Tool

In this section, we present the time-sharing-like tool mentioned previously. In particular, using this tool, combined with the two points from Corollary 2, allows us fully characterize the part of the region in which we achieve the capacity.

Furthermore, using the following Lemma, we can extend the schemes presented in [1] and, generally, show that if any two points are achievable, (practically) any linear combination between such two points is also achievable.

Lemma 2. *For SWDF (see Definition 9): if an* $(n_1, n_2, n_3, k_1, k_2, T)_{\mathbb{F}}$ *and an* $(n'_1, n'_2, n'_3, k'_1, k'_2, T)_{\mathbb{F}}$ *streaming codes are* (N_1, N_2, N_3)*-achievable, there exists an* (N_1, N_2, N_3)*-achievable* $(An_1 + Bn'_1, An_2 + Bn'_2, An_3 + Bn'_3, Ak_1 + Bk'_1, Ak_2 + Bk'_2, T)_{\mathbb{F}}$ *streaming code* $\forall A, B \in \mathbb{Z}$.

1) Concatenated Symbol-wise Decode-and-forward: Recall that, as shown in [1], the following streaming codes are (N_1, N_2, N_3)-achievable: $(T+1-N_3, 0, T+1-N_1, T+1-N_1-N_3, 0)_{\mathbb{F}}$ and $(0, T+1-N_3, T+1-N_2, 0, T+1-N_2-N_3)_{\mathbb{F}}$. The CSWDF simply uses 2 with these codes. In particular, the following pair of rates is, to the best of our knowledge, the best achievable rate for R_1 such that $R_2 = C(T-N_3, N_2)$ and can be obtained using CSWDF.

Lemma 3. *For* $R_2 = C(T-N_3, N_2)$*, the following* R_1 *is achievable*

$$R_1 = \frac{(T+1-N_3-N_1)(N_2-N_3)}{(T+1-N_1)(T+1-N_3)} \tag{4}$$

using diagonal interleaving MDS code.

2) Concatenated Message-wise Decode-and-forward: Similar to the CSWDF, we can apply Lemma 2 to the message-wise decode-and-forward scheme[1] presented in [1]. A full description of the scheme can be found in [16].

VI. Results

In this section, we present the rate region for two different scenarios. In both cases, the curves are labeled according to the scheme presented in each respective section previously. The curve labeled as “Time Sharing” in Fig 2 represents the best achievable region for which we have closed form expressions.

We also present the results for the OB-SWDF scheme, which is fully described in [16]. Briefly, this scheme optimizes over the rate of the point-to-point code used in the second hop, in contrast to the fixed rate used by FB-SWDF. By doing that, we are able to improve the achievable rate in the scenarios where FB-SWDF is unable to achieve the upper bound.

In the first scenario, we have a small T, such that the condition in Corollary 2 is not met. This is presented in Fig. 2. Note that, in this case, no scheme is able to achieve the sumrate. By slightly increasing T, we are able to achieve the sumrate capacity in a noticeable part of the capacity region, which is shown in Fig. 3. Note that, in this case, the message-wise scheme is unable to achieve even the single-user capacity for R_2, which is also shown in [1].

Furthermore, in both cases, it can be seen that CSWDF is able to achieve the optimal point for maximal R_2. This has been observed in all settings we have experimented, and, although it remains to be proven, it suggests that the “Time Sharing” scheme represents a good achievable rate, for which we have closed form expressions. In fact, as in Fig. 3, it can achieve the same performance as OB-SWDF.

Although our schemes are unable to always achieve the sumrate, and unable to achieve the entire rate region, comparing them to the alternative—CMWDF—should show that the proposed schemes are significantly superior.

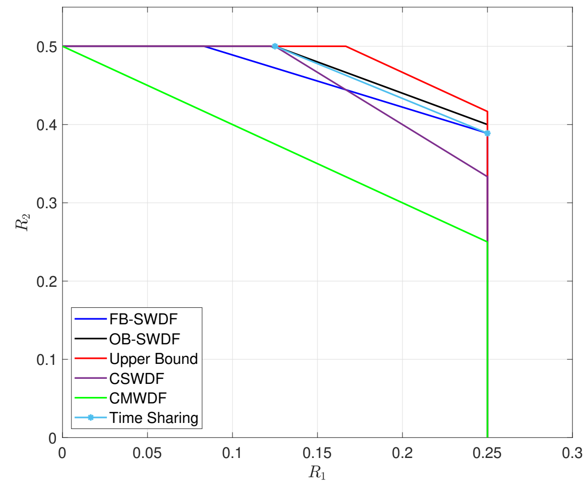

Fig. 2: Rate region for $N_1 = 3, N_2 = 2, N_1 = 1, T = 4$.

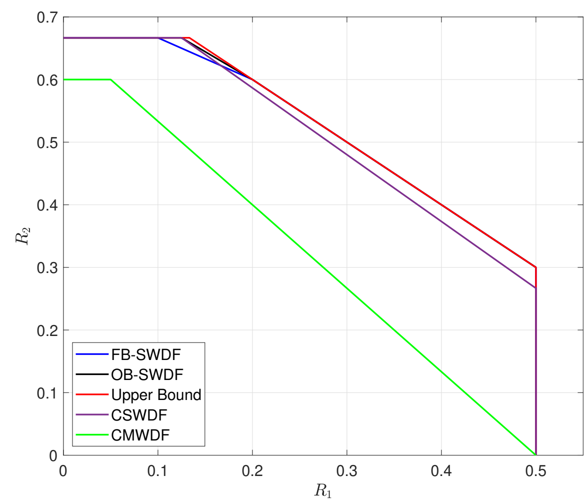

Fig. 3: Rate region for $N_1 = 3, N_2 = 2, N_1 = 1, T = 6$.

[1]Recall that message-wise decode-and-forward is a particular case of SWDF, thus Lemma 2 can be applied.

References

[1] S. L. Fong, A. Khisti, B. Li, W. Tan, X. Zhu, and J. Apostolopoulos, "Optimal streaming erasure codes over the three-node relay network," in *2019 IEEE International Symposium on Information Theory (ISIT)*, July 2019, pp. 3077–3081.

[2] E. Martinian and C.-E. Sundberg, "Burst erasure correction codes with low decoding delay," *IEEE Transactions on Information theory*, vol. 50, no. 10, pp. 2494–2502, 2004.

[3] D. Leong and T. Ho, "Erasure coding for real-time streaming," in *2012 IEEE International Symposium on Information Theory Proceedings*, 2012, pp. 289–293.

[4] A. Badr, A. Khisti, W.-T. Tan, and J. Apostolopoulos, "Streaming codes for channels with burst and isolated erasures," in *2013 Proceedings IEEE INFOCOM*. IEEE, 2013, pp. 2850–2858.

[5] T. Ho, M. Medard, J. Shi, M. Effros, and D. R. Karger, "On randomized network coding," in *Proceedings of the Annual Allerton Conference on Communication Control and Computing*, vol. 41, no. 1. Citeseer, 2003, pp. 11–20.

[6] G. Joshi, Y. Kochman, and G. W. Wornell, "On playback delay in streaming communication," in *2012 IEEE International Symposium on Information Theory Proceedings*, 2012, pp. 2856–2860.

[7] M. Karzand, D. J. Leith, J. Cloud, and M. Médard, "Design of FEC for low delay in 5G," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 8, pp. 1783–1793, 2017.

[8] A. Badr, P. Patil, A. Khisti, W.-T. Tan, and J. Apostolopoulos, "Layered constructions for low-delay streaming codes," *IEEE Transactions on Information Theory*, vol. 63, no. 1, pp. 111–141, 2017.

[9] A. Badr, A. Khisti, W.-t. Tan, X. Zhu, and J. Apostolopoulos, "FEC for VoIP using dual-delay streaming codes," in *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*. IEEE, 2017, pp. 1–9.

[10] M. Rudow and K. V. Rashmi, "Streaming codes for variable-size arrivals," in *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2018, pp. 733–740.

[11] M. N. Krishnan and P. V. Kumar, "Rate-optimal streaming codes for channels with burst and isolated erasures," in *2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2018, pp. 1809–1813.

[12] S. L. Fong, A. Khisti, B. Li, W. Tan, X. Zhu, and J. Apostolopoulos, "Optimal streaming codes for channels with burst and arbitrary erasures," *IEEE Transactions on Information Theory*, vol. 65, no. 7, pp. 4274–4292, 2019.

[13] E. Domanovitz, S. L. Fong, and A. Khisti, "An explicit rate-optimal streaming code for channels with burst and arbitrary erasures," *arXiv preprint arXiv:1904.06212*, 2019.

[14] M. N. Krishnan, D. Shukla, and P. V. Kumar, "Low field-size, rate-optimal streaming codes for channels with burst and random erasures," *IEEE Transactions on Information Theory*, pp. 1–1, 2020.

[15] E. Domanovitz, A. Khisti, W.-T. Tan, X. Zhu, and J. Apostolopoulos, "Streaming erasure codes over multi-hop relay network," *CoRR*, vol. 2006.05951, 2020. [Online]. Available: https://arxiv.org/abs/2006.05951

[16] G. K. Facenda, E. Domanovitz, W. T. A. Khisti, and J. Apostolopoulos, "Streaming erasure codes over the multiple access relay channel," *arXiv preprint arXiv:2101.11117*, 2021.

[17] G. Kramer and A. J. van Wijngaarden, "On the white gaussian multiple-access relay channel," in *2000 IEEE International Symposium on Information Theory (Cat. No.00CH37060)*, 2000, pp. 40–.

[18] A. Badr, D. Lui, A. Khisti, W. Tan, X. Zhu, and J. Apostolopoulos, "Multiplexed coding for multiple streams with different decoding delays," *IEEE Transactions on Information Theory*, vol. 64, no. 6, pp. 4365–4378, June 2018.