Simple Streaming Codes for Reliable, Low-Latency Communication

M. Nikhil Krishnan*, Vinayak Ramkumar*, Myna Vajha*, P. Vijay Kumar*
Electrical Communication Engineering, Indian Institute of Science, Bangalore, 560012
email: {nikhilkrishnan.m, vinram93, mynaramana, pvk1729}@gmail.com

***Abstract*—Streaming codes offer reliable recovery under decoding-delay constraint τ, of packets transmitted over a burst-and-random-erasure channel. Prior rate-optimal code constructions had field size quadratic in τ and employed diagonal embedding of a scalar block code of length n within the packet stream. It is shown here that staggered diagonal embedding (SDE) under which the n code symbols are dispersed across a span of $N \geq n$ successive packets leads to a simpler, low-complexity construction of rate-optimal streaming codes having linear field size. The limits of the SDE approach under the restriction $N \leq \tau+1$ are explored. Some binary streaming codes that are rate-optimal under this restriction are identified.**

Keywords: streaming codes, low-latency communication, burst and random erasure correction, packet-level FEC.

I. Introduction

Ensuring reliable, low-latency communication is key to enabling next-generation applications such as disaster recovery, industrial automation, interactive streaming, multi-player gaming, telesurgery, vehicular safety and virtual reality. Not surprisingly, Ultra-Reliable, Low-Latency Communication (URLLC) is one of the three core focus areas of 5G. The ITU-R report [1] sets a latency requirement of 1 millisecond for URLLC, where latency is measured as the time elapsed between transmission of a packet from the source and its recovery at the receiver. At the same time, it is also desired to keep the failure probability to acceptably low levels. It is challenging to meet the tight latency constraints while employing ARQ-based schemes on account of the large, inherent round-trip delays. On the other hand, ensuring reliability through the blind re-transmission of data is wasteful of resources. For these reasons, a viable alternative is the use of Forward Error Correction (FEC). Streaming codes represent a packet-level FEC scheme for ensuring reliable, low-latency communication.

Research on streaming codes began with the early work of Martinian-Sundberg [2] and Martinian-Trott [3] in which a packet-expansion encoding framework was introduced. Let $\underline{u}(t) \in \mathbb{F}_q^k$, denote the message packet at time t, where $\mathbb{F}_q$ denotes a finite field of size q. The encoder generates a coded packet $\underline{x}(t) \in \mathbb{F}_q^n$, $\underline{x}(t)^T = \left[\underline{u}(t)^T \ \underline{p}(t)^T\right]$, that is then transmitted across the channel. The encoder is assumed to be causal and hence the parity packet $\underline{p}(t)$ at time t is a function only of the current and prior message packets. Constructions of streaming codes that can protect against a burst erasure of length $\leq b$ while incurring a decoding delay no larger than τ can be found in [2], [3]. The decoding-delay constraint requires recovery of packet $\underline{u}(t)$ by time $t+\tau$. In [4], Badr et al. extend this approach to a more general, delay-constrained sliding-window (SW) channel model that can cause both burst and random erasures. This channel model may be regarded as a tractable approximation of the commonly-accepted Gilbert-Elliott (GE) channel model. Under the SW channel model, within any sliding window of time duration w, the channel can introduce either a burst erasure of length $\leq b$ or else, at most a random erasures. Further, decoding must take place within delay constraint τ. Clearly, we can assume that $a \leq b \leq \tau$. As is explained in [4], [5], one can also set $w = \tau+1$ without loss of generality. Thus the SW channel model is characterized by the parameter set $\{a, b, \tau\}$. We will refer to codes that enable recovery from all the permissible erasure patterns of such a SW channel as (a, b, τ) streaming codes. The paper [4] also contains the following upper bound on the rate R of an (a, b, τ) streaming code:

$$R \quad \leq \quad \frac{\tau+1-a}{\tau+1-a+b} \ \triangleq \ R_{\text{opt}}. \qquad (1)$$

The same paper also includes near-rate-optimal constructions of (a, b, τ) streaming codes for all parameter sets. In [6], the authors present a rate-optimal construction for the specific case $\tau+1 = a+b$. The first rate-optimal families of streaming codes for all $\{a, b, \tau\}$ parameters appeared in the independent works [7], [8]. These code families required operation over a finite field having size exponential in τ. In a subsequent paper [5], the authors provide a general construction, valid for all parameter sets, with a field size requirement that is quadratic in τ. The same paper also contains 4 additional constructions with linear field size, but only for the rather restricted cases: (i) $b = a+1$, (ii) $(\tau+a+1) \geq 2b \geq 4a$, (iii) $a \mid b \mid (\tau+1-a)$ and (iv) $b = 2a-1$ and $b \mid (\tau+2-a)$, where the notation $x \mid y$ denotes that x divides y. In the recent work [9], the authors provide an explicit code construction, which once again, requires a field size that is quadratic in τ. In each of [5], [7]–[9], the authors obtain (a, b, τ) streaming codes by employing the technique introduced in [3], of diagonally embedding the code symbols of a scalar block code $\mathcal{C}$ of block length n within the packet stream.

The contributions of the present paper are the following.

* indicates equal contribution.

P. Vijay Kumar is also a Visiting Professor at the University of Southern California. This research is supported in part by the J C Bose National Fellowship JCB/2017/000017 and in part by the NetApp University Research Fund SVCF-0002. Myna Vajha would like to acknowledge the support of Visvesvaraya PhD Scheme. M. Nikhil Krishnan and Myna Vajha would like to acknowledge the support of Qualcomm Innovation Fellowship, India.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCOMM.2019.2956500, IEEE Communications Letters

(a) It is shown how replacing the earlier diagonal embedding approach, by staggered-diagonal embedding (SDE), reduces the burden of erasure recovery placed on the underlying block code $\mathcal{C}$, leading to simpler streaming-code constructions having smaller block length and reduced field size. Under SDE, the n code symbols of $\mathcal{C}$ are dispersed across a span of N packets and we will refer to N as the dispersion-span parameter.

(b) The SDE approach yields (a, b, τ) streaming codes for all parameters. These codes require a linear field size less than $\tau + 1$, and have decoding complexity equivalent to that of decoding an MDS code of block length at most $\tau + 1$. Furthermore, when either $b \mid (\tau + 1 - a)$ or else $b = a$, the resultant codes can be shown to be rate-optimal with respect to (1).

(c) The limits of the SDE approach while restricting the dispersion-span parameter N to be no larger than $\tau + 1$ are explored. Limiting N leads to a further reduction in the requirements placed on scalar block code $\mathcal{C}$, thereby simplifying code design. The maximum achievable rate R_{max} under the restriction $N \leq \tau + 1$ is determined. It is shown that this rate is always achievable using an MDS code of redundancy a. Of significant practical interest, is the fact that in some instances, this maximum rate R_{max} is also achievable by a binary streaming code.

In Section II, an example is presented that illustrates the advantage of the SDE approach over diagonal-embedding. The SDE approach is formally introduced in Section III. Streaming code constructions that make use of the SDE approach are provided in Section IV. In Section V, we explore the limits of the SDE approach under the constraint $N \leq \tau + 1$ on the dispersion-span. A comparative performance evaluation of different streaming codes over the GE channel is presented in Section VI. Section VII draws conclusions. Throughout, we use $[a : b]$ to denote the set $\{a, a + 1, \ldots, b - 1, b\}$.

II. A Motivating Example

In this section, both SDE and diagonal-embedding approaches are employed to construct $(a = 4,\ b = 6,\ \tau = 9)$ streaming codes achieving $R_{\text{opt}} = \frac{1}{2}$. While the known rate-optimal, diagonal-embedding constructions require field size $q \geq 81$ and packet length $n = 12$, these requirements drop to $q < 8$ and $n = 8$ under the SDE approach.

A. Diagonal-Embedding Approach

For any $i \in [0 : n - 1]$, let $\mathcal{E}_i \subseteq [i : \min\{i + \tau, n - 1\}]$, denote an erasure set composed of at most b consecutive coordinates or else, a arbitrarily chosen coordinates from $[i : \min\{i + \tau, n - 1\}]$. Diagonal-embedding approach calls for an $[n, k]$ scalar block code $\mathcal{C}$ having the property that for any erasure set $\mathcal{E}_i$ such that $i \in \mathcal{E}_i$, the ith code symbol c_i, can be recovered from knowledge of the prior code symbols $\{c_j\}_{j \in [0:i-1]}$ and the subsequent code symbols $\{c_{i+\ell}\}_{\ell \in [1:\min\{\tau, n-i-1\}] \setminus \mathcal{E}_i}$. The symbols of codewords in $\mathcal{C}$ are then placed along the diagonal in packet stream as shown by the blue-shaded rectangles appearing in Fig. 1. Note from Fig. 1, that an erasure burst of length 6 (indicated by the red dotted line) erases 6 symbols of the codeword in blue. This implies that the number of parity symbols $(n - k)$ must be at least $b = 6$. It follows then from the rate bound in (1), that for the resultant streaming code to be rate-optimal, the block-length n should be at least $\tau + 1 - a + b = 12$. Furthermore, the smallest-known field size q required to construct a $[12, 6]$ scalar code with these properties is $q \geq 81$ (based on the best constructions known to date and contained in [5] or [9]).

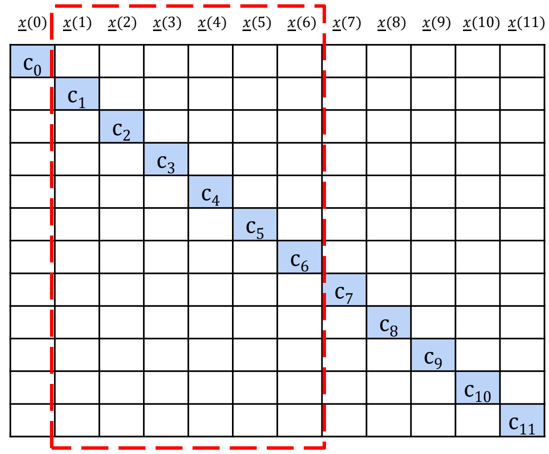

Figure 1: Diagonal embedding of a codeword belonging to a $[12, 6]$ scalar block code $\mathcal{C}$ constructed in either [5] or [9]. Each column indicates a coded packet. Here, the red dotted line depicts a burst erasure of length $b = 6$ and c_1 needs to be recoverable from symbols $\{c_0, c_7, c_8, c_9, c_{10}\}$.

B. Staggered-Diagonal Embedding Approach

Under the SDE approach, for the same parameter set $(a = 4,\ b = 6,\ \tau = 9)$, the construction begins by identifying an $[8, 4]$ MDS code $\mathcal{C}$ which we will refer to as the base code, and which requires a field size < 8. Let $\underline{x}(t)^T = [\underline{u}(t)^T\ \underline{p}(t)^T]$ denote the coded packet of length $n = 8$ at time t where $\underline{u}(t) = [u_0(t)\ u_1(t)\ u_2(t)\ u_3(t)]^T$ denotes the message symbols at time t and $\underline{p}(t) = [p_0(t)\ p_1(t)\ p_2(t)\ p_3(t)]^T$, the parity symbols. The parity symbols $\{p_i(t)\}$ are designed to be a function of message symbols belonging to prior packets in such a way that for every t, the collection of symbols

$$(u_0(t), u_1(t+1), u_2(t+2), u_2(t+3), p_0(t+6), p_1(t+7), p_2(t+8), p_3(t+9))$$

forms a codeword in the base code $\mathcal{C}$. Note that there are two gaps, namely $\{t + 4, t + 5\}$, in the associated time series. As shown in Fig. 2, it is the presence of these gaps that leads to staggered-diagonal embedding, within the packet stream, of a codeword from the $[8, 4]$ MDS code.

It remains to explain how this arrangement results in an $(a = 4, b = 6, \tau = 9)$ streaming code. Consider any window of length $\tau + 1 = 10$ packets and suppose first, that there are 4 packet erasures, in arbitrary locations, within this window. This results in the erasure of at most 4 code symbols in arbitrary locations, within any of the codewords, drawn from $\mathcal{C}$ involved in that window. Next, consider the case when the window of length 10 packets contains a burst erasure of length 6 packets (indicated by the red dotted line). It is easy to see that on account of the staggered-diagonal arrangement, once again, at most four symbols are erased from any of the codewords drawn from $\mathcal{C}$ involved in that window. Since the MDS code

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCOMM.2019.2956500, IEEE Communications Letters

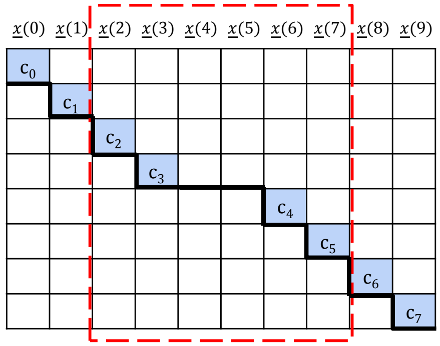

Figure 2: Illustrating the staggered-diagonal embedding of a codeword in the $[8,4]$ MDS code $\mathcal{C}$ within the packet stream. Here also, each column indicates a coded packet.

has four parities, it can recover from any four erasures, thus guaranteeing full erasure recovery in both instances. Note that under this arrangement, any codeword is dispersed across only 10 consecutive packets and hence the decoding-delay constraint of $\tau = 9$ is trivially met.

III. Staggered-Diagonal Embedding

We now present the SDE approach in general. Let $\mathcal{C}$ be an $[n,k]$ linear code in systematic form, with the first k code symbols corresponding to the message. Let $N \geq n$ be an integer and let S be a subset of $[0:N-1]$ of size n:

$$S = \{i_0, i_1, \ldots, i_{n-1}\} \subseteq [0:N-1],$$

where $0 = i_0 < i_1 < \cdots < i_{n-1} = N-1$. Let J denote the complement $J = [0:N-1] \setminus S$ of S in $[0:N-1]$. Let $x_i(t)$ denote the ith component of the coded packet $\underline{x}(t)$. Then for every t, the collection of symbols

$$\Big(x_0(t+i_0), x_1(t+i_1), \cdots, x_{n-1}(t+i_{n-1})\Big)$$

forms a codeword in code $\mathcal{C}$. We will refer to S as the placement set as it determines the placement of code symbols of $\mathcal{C}$ within the packet stream and J as the gap set as it identifies gaps in occupancy within the packet stream, of code symbols from a given codeword in $\mathcal{C}$. The packet-level code thus constructed through SDE, has rate $\frac{k}{n}$. It is easily seen that when $J = \phi$, we have $N = n$, and the resulting embedding reduces to diagonal embedding.

A. The Base and Dispersed Codes

We will refer to $\mathcal{C}$ as the base code. For ease in description and analysis, we introduce a second code, which we call the dispersed code. We begin by associating with every codeword $\underline{c} = (c_0, c_1, \ldots, c_{n-1})$ in the base code $\mathcal{C}$, a dispersed vector $\underline{d}(\underline{c}) = (d_0, d_1, \ldots, d_{N-1})$, given by $d_{i_j} = c_j$, for all $j \in [0:n-1]$ and $d_l = 0$ for all $\ell \in J$. We define $\mathcal{D}(\mathcal{C}, S) \triangleq \{\, \underline{d}(\underline{c}) \mid \underline{c} \in \mathcal{C} \,\}$. Clearly, $\mathcal{D}(\mathcal{C}, S)$ is a linear code of block length N. We will refer to $\mathcal{D}(\mathcal{C}, S)$ as the dispersed code associated to base code $\mathcal{C}$ and placement set S and will refer to N as the dispersion-span parameter. For simplicity, we will write $\mathcal{D}$ in place of $\mathcal{D}(\mathcal{C}, S)$ whenever $\mathcal{C}$ and S are either clear from the context or else irrelevant to the discussion. Although at times, we will abuse terminology and make reference to the SDE of a dispersed code $\mathcal{D}(\mathcal{C}, S)$, such a reference should be reinterpreted as SDE of the underlying base code $\mathcal{C}$. In the example considered in Section II, the base code $\mathcal{C}$ is an $[8,4]$ MDS code, $S = \{0,1,2,3,6,7,8,9\}$ and $J = \{4,5\}$ (see Fig. 3).

d_0	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9
c_0	c_1	c_2	c_3	0	0	c_4	c_5	c_6	c_7

Figure 3: Codeword in the dispersed code $\mathcal{D}(\mathcal{C}, S)$ associated to $[8,4]$ MDS code $\mathcal{C}$ and $S = \{0,1,2,3,6,7,8,9\}$.

Throughout the remainder of the paper, we will restrict our attention to the case when the dispersion-span parameter N satisfies $N \leq \tau + 1$. This limitation on block length N of the dispersed code serves two purposes. Firstly, it ensures that under the SDE approach, the decoding delay constraint is trivially satisfied. Secondly, it tends to reduce the length n of a packet under the SDE approach. The utility of introducing the notion of a dispersed code can be seen in Theorem III.1.

Theorem III.1. *Let $\mathcal{C}$ be an $[n, k, d_{min}]$ code and N an integer such that $n \leq N \leq \tau+1$. Let $S \subseteq [0:N-1]$ with $|S| = n$ and $\mathcal{D}$ be the associated dispersed code. Then SDE of the dispersed code will result in an (a, b, τ) streaming code if and only if the dispersed code $\mathcal{D}$ has $d_{min}(\mathcal{D}) \geq a+1$ and can recover from any burst of b erasures. Further, the requirement $d_{min}(\mathcal{D}) \geq a+1$ can be replaced by the requirement $d_{min}(\mathcal{C}) \geq a+1$.*

Proof. The limitation $N \leq \tau + 1$ on block length of the dispersed code ensures that the decoding-delay constraint is trivially satisfied. The random and burst-erasure properties of the streaming code are easily verified to be satisfied. The last property follows since both $\mathcal{C}$ and $\mathcal{D}$ clearly have the same value of minimum distance. ☐

IV. Code Constructions Under the SDE Approach

We present constructions in this section, of dispersed codes that meet the random and burst-erasure conditions laid out in Theorem III.1, thereby yielding (a, b, τ) streaming codes. Furthermore, all the constructions meet the dispersion-span constraint $N \leq \tau + 1$. Let $\tau + 1 = mb + \delta_1$, with $0 \leq \delta_1 < b$. Then from (1), the optimal rate R_{opt} takes on the form $R_{\text{opt}} = \frac{mb+\delta_1-a}{(m+1)b+\delta_1-a}$.

A. The MDS-Base-Code Construction for All Parameters

We set $\delta_2 = \min\{\delta_1, a\}$ and pick the base code $\mathcal{C}$ to be an $[n = ma + \delta_2,\ k = (m-1)a + \delta_2]$ MDS code of redundancy $n - k = a$. We set $N \triangleq mb + \delta_2$ and note that $N \leq \tau + 1$. We select placement set

$$S = \bigcup_{i=0}^{m-1} [ib : ib + a - 1] \ \cup\ [mb : mb + \delta_2 - 1].$$

Clearly, $d_{\min}(\mathcal{C}) = a + 1$ and a field size $(n-1)$ suffices for the construction of base code $\mathcal{C}$. It can be seen that the erasure of any successive b coordinates in the dispersed code $\mathcal{D}(\mathcal{C}, S)$ corresponds to the erasure of exactly a code symbols belonging to the base code $\mathcal{C}$. The MDS nature of $\mathcal{C}$ guarantees recovery from these a erasures. Thus the conditions laid down in Theorem III.1 are met and SDE of $\mathcal{D}(\mathcal{C}, S)$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCOMM.2019.2956500, IEEE Communications Letters

will thus result in an (a, b, τ) streaming code having rate $R_{\text{MDS}} = \frac{k}{n} = \frac{(m-1)a+\delta_2}{ma+\delta_2}$. As Table I shows, the MDS-base-code construction is rate-optimal when either $\delta_1 = a$, i.e., $b|(\tau + 1 - a)$ or else $b = a$. Table II presents some example parameters.

Remark IV.1. Construction C in [5], providing rate-optimal codes for the case $a|b|(\tau+1-a)$, may be viewed as a special case of the MDS-base-code construction.

Condition	R_{MDS}	R_{opt}
$\delta_1 < a$	$\frac{m-(\frac{a-\delta_1}{a})}{m+1-(\frac{a-\delta_1}{a})}$	$\frac{m-(\frac{a-\delta_1}{b})}{m+1-(\frac{a-\delta_1}{b})}$
$\delta_1 = a$	$\frac{m}{m+1}$	$\frac{m}{m+1}$
$\delta_1 > a$	$\frac{m}{m+1}$	$\frac{m+(\frac{\delta_1-a}{b})}{m+1-(\frac{\delta_1-a}{b})}$

Table I: Comparing R_{MDS} with R_{opt}.

a	b	τ	m	δ_1	N	n	k	R_{MDS}	R_{opt}
2	3	3	1	1	4	3	1	0.333	0.4
2	3	4	1	2	5	4	2	0.5	0.5
2	3	5	2	0	6	4	2	0.5	0.571
3	5	11	2	2	12	8	5	0.625	0.6429
3	5	12	2	3	13	9	6	0.666	0.666
3	5	13	2	4	13	9	6	0.666	0.6875

Table II: Example parameters: MDS-base-code construction.

B. The Binary, MBER-Base-Code Construction

When $\delta_1 > a$, it is possible in some cases, to construct binary streaming codes having the same rate $\frac{m}{m+1}$ as that achieved by the MDS-base-code construction, by employing a binary code as the base code. Furthermore, as shown in Section V, these codes achieve the maximum rate $R_{\max}$ possible under the SDE approach, when the dispersion-span constraint $N \leq \tau + 1$ is imposed.

If an $[n = (m+1)r,\ k = mr,\ d_{\min} \geq a+1]$ binary cyclic code exists for some $r \in [a+1 : \delta_1]$, we choose this as the base code $\mathcal{C}$ and set $S = \bigcup_{i=0}^{m} [ib : ib+r-1]$. From the placement set structure, it is easy to see that any b consecutive coordinates of $\mathcal{D}(\mathcal{C}, S)$ will contain exactly r consecutive code symbols of $\mathcal{C}$. Any $[n, k]$ cyclic code can recover from a burst erasure of length $(n-k)$ and hence $\mathcal{C}$ can handle a burst erasure of length r. Therefore, by Theorem III.1, we have constructed an (a, b, τ) streaming code of rate $R = \frac{k}{n} = \frac{m}{m+1}$. Since $\delta_1 > a$, we have $R = \frac{m}{m+1} < R_{\text{opt}} < \frac{m+1}{m+2}$. There exists non-cyclic binary codes that can correct any burst erasure of length $(n-k)$ and which can thus equally effectively, be chosen as the base code. For example, an $[n = 8,\ k = 4,\ d_{\min} = 4]$ extended Hamming code can correct any burst erasure of length $(n-k) = 4$. Since no code having redundancy $(n-k)$ can correct a burst erasure of length $> (n-k)$, these codes may be regarded as binary, maximal-burst-erasure-recovery (MBER) codes. Table III presents some example parameters of binary streaming codes arising from the use of an existing binary MBER code as the base code.

C. The Repetition-Base-Code Construction

Here we use the $[a+1, 1]$ repetition code as the base code $\mathcal{C}$ and set $S = [0 : a-1] \cup \{\tau\}$. The code $\mathcal{C}$ clearly has

a	b	τ	m	δ_1	N	n	k	$R = R_{\max}$	R_{opt}
3	6	9	1	4	10	8	4	0.5	0.538
3	7	10	1	4	11	8	4	0.5	0.533
3	6	16	2	5	17	15	10	0.666	0.7
3	7	19	2	6	19	15	10	0.666	0.708
3	8	21	2	6	21	15	10	0.666	0.703
3	9	22	2	5	23	15	10	0.666	0.689

Table III: This table provides examples for $\delta_1 > a$, where the MBER-base-code construction results in binary streaming codes achieving rate $R_{\max}$ (see Definition V.1). The $[8, 4, 4]$ code above is an extended Hamming code, whereas the $[15, 10, 4]$ code is a BCH code.

$d_{\min} = a+1$ and no burst erasure of length b can erase both the first and the last symbols of the dispersed code $\mathcal{D}(\mathcal{C}, S)$ since $b \leq \tau$. Thus, SDE of $\mathcal{D}(\mathcal{C}, S)$ will result in an (a, b, τ) streaming code of rate $R = \frac{1}{a+1}$ for any $a \leq b \leq \tau$.

Consider the class of streaming codes where no computation is allowed for erasure recovery. This forces the parity symbols $\{p_j(t)\}$ of the streaming code to be simple repetitions of the message symbols, i.e., $p_j(t) = u_i(t-\ell)$, some $\ell \geq 0$. Since the channel model allows a random erasures within a window of length $(\tau+1)$, any message symbol $u_i(t)$ must appear at least $a+1$ times within a window of $(\tau+1)$ packets starting at t, to ensure guaranteed recovery from these a erasures under decoding-delay constraint τ. It follows that the rate of such an FEC scheme cannot possibly exceed $\frac{1}{a+1}$. Thus in this setting, the repetition-base-code construction is rate-optimal, achieving rate $R = \frac{1}{a+1}$.

V. Maximum Rate Achievable Under the SDE Approach

In this section, we determine the quantity $R_{\max}$ defined as follows.

Definition V.1. *Let R_{max} be the maximum possible rate of an (a, b, τ) streaming code constructed using the SDE approach, under the dispersion-span constraint $N \leq \tau + 1$.*

Theorem V.1. *Let $\mathcal{C}$ and $\mathcal{D}$ be the $[n, k]$ base code and $[N, k]$ dispersed codes respectively, employed in the construction of an (a, b, τ) streaming code under the SDE approach, with $N \leq \tau + 1$. Let $\tau + 1 = mb + \delta_1$ for some integer $m \geq 1$, and $0 \leq \delta_1 < b$. Then*

$$R_{max} = \begin{cases} \frac{m}{m+1}, & \delta_1 \geq a \\ \frac{m-(\frac{a-\delta_1}{a})}{m+1-(\frac{a-\delta_1}{a})}, & \delta_1 < a. \end{cases} \quad (2)$$

The maximum rate is always achieved by selecting the base code to be an $[n, k]$ MDS code with $n - k = a$. Furthermore, the maximum rate R_{max} coincides with the optimal rate R_{opt} iff $\delta_1 = a$, i.e., $b|(\tau + 1 - a)$ or else $a = b$.

Proof. Let $r \triangleq (n-k)$ denote the redundancy of the base code. For the dispersed code $\mathcal{D}$ to be able to correct b burst erasures, it is necessary that any b-length window contain at most r code symbols of $\mathcal{C}$. This is because it is impossible for a linear code $\mathcal{C}$ of redundancy r to recover from more than r erasures. It follows from this and the dispersion-span constraint $N \leq \tau+1 = mb + \delta_1$, that the block length n of the base code $\mathcal{C}$ is limited by the upper bound $n \leq mr + \delta_2$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCOMM.2019.2956500, IEEE Communications Letters

where $\delta_2 = \min\{\delta_1, r\}$. This leads to the following upper bound on the rate $R = k/n$ of $\mathcal{C}$:

$$R \leq \frac{(m-1)r + \delta_2}{mr + \delta_2} = \min\left\{\frac{m}{m+1}, \frac{m-1+\frac{\delta_1}{r}}{m+\frac{\delta_1}{r}}\right\}$$
$$\leq \min\left\{\frac{m}{m+1}, \frac{m-1+\frac{\delta_1}{a}}{m+\frac{\delta_1}{a}}\right\},$$

since $a \leq d_{\min}(\mathcal{C}) - 1 \leq r$. This rate is achieved however, by the MDS-base-code construction (see Table I) for both cases $\delta_1 < a$ and $\delta_1 \geq a$. The values for $R_{\max}$ specified in equation (2) follow. Also, since

$$R_{\text{opt}} = \frac{\tau+1-a}{\tau+1-a+b} = \frac{m+\frac{\delta_1-a}{b}}{m+1+\frac{\delta_1-a}{b}},$$

it follows that $R_{\max} = R_{\text{opt}}$ iff $\delta_1 = a$ or else $a = b$. □

Remark V.1. The maximum rate $R_{\max} = \frac{m}{m+1}$ for the case $\delta_1 > a$, can also be achieved by the binary, MBER-base-code construction, if for some $r \in [a+1, \delta_1]$ an $[n = (m+1)r,\ k = mr]$, binary, MBER code with $d_{\min} \geq a+1$ exists. Examples appear in Table III.

VI. Numerical Evaluation

In this section, we compare the performance over the widely-employed Gilbert-Elliott (GE) channel model, of codes constructed under the SDE approach against prior coding schemes. The GE channel consists of a good state G and a bad state B with transition probabilities between the two as shown in Fig 4. It is assumed here that in the good state, the GE channel behaves as a binary erasure channel with erasure probability ϵ and in the bad state, as a binary erasure channel with erasure probability 1.

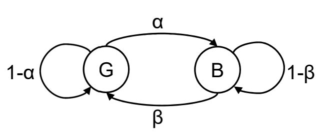

Figure 4: Gilbert-Elliott channel

Our simulations involved a GE channel with $\alpha = 10^{-4}, \beta = 0.6$ and ϵ ranging from 0.001 to 0.02. All the codes were chosen to have rate $\frac{1}{2}$ and decoding-delay $\tau = 9$. In conjunction with diagonal embedding, we evaluate the following constructions:

(a) Martinian-Trott construction [3]: Binary $[n = 18, k = 9]$ code, which can correct burst erasures of length 9.
(b) Krishnan et al. construction [5]: $(a = 3, b = 7, \tau = 9)$ streaming code with $[n = 14, k = 7]$ over $\mathbb{F}_{2^8}$.
(c) Domanovitz et al. construction [9]: $(a = 3, b = 7, \tau = 9)$ streaming code with $[n = 14, k = 7]$ over $\mathbb{F}_{2^8}$.
(d) $(a = 5, b = 5, \tau = 9)$ streaming code constructed using an $[n = 10, k = 5]$ MDS code over $\mathbb{F}_{2^4}$.

With regard to SDE, we use the following constructions:

(i) MDS-base-code construction: $(a = 3, b = 7, \tau = 9)$ streaming code with an $[n = 6, k = 3]$ MDS base code over $\mathbb{F}_{2^3}$ and placement set $S = \{0, 1, 2, 7, 8, 9\}$.
(ii) Binary MBER-base-code construction: $(a = 3, b = 6, \tau = 9)$ streaming code with an $[n = 8, k = 4, d_{\min} = 4]$ extended binary Hamming code as base code and placement set $S = \{0, 1, 2, 3, 6, 7, 8, 9\}$.

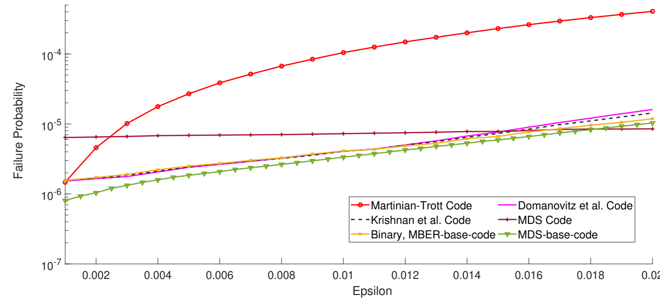

Figure 5: Numerical evaluation over GE channel with $\alpha = 10^{-4}, \beta = 0.6$.

As can be seen from Fig 5, codes constructed using the SDE approach compare favorably with prior constructions.

VII. Conclusions

A staggered-diagonal embedding (SDE) approach was introduced here, that leads to simpler, streaming-code constructions requiring linear finite field size and smaller packet lengths, thereby resulting in substantially reduced encoding and decoding complexity. For the case when $b|(\tau+1-a)$, the codes are rate-optimal. The maximum rate $R_{\max}$ achievable using the SDE approach under the dispersion-span constraint $N \leq \tau+1$ is determined and it is shown that this rate is always achievable using MDS base codes. Interestingly, in many instances, $R_{\max}$ is achievable using a binary base code.

References

[1] ITU-R, *Minimum requirements related to technical performance for IMT-2020 radio interface(s)*. Report M.2410-0, Nov. 2017.
[2] E. Martinian and C. W. Sundberg, "Burst erasure correction codes with low decoding delay," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2494–2502, 2004.
[3] E. Martinian and M. Trott, "Delay-Optimal Burst Erasure Code Construction," in *Proc. Int. Symp. Inf. Theory, Nice, France, June 24-29, 2007*. IEEE, pp. 1006–1010.
[4] A. Badr, P. Patil, A. Khisti, W. Tan, and J. G. Apostolopoulos, "Layered Constructions for Low-Delay Streaming Codes," *IEEE Trans. Inf. Theory*, vol. 63, no. 1, pp. 111–141, 2017.
[5] M. N. Krishnan, D. Shukla, and P. V. Kumar, "Low Field-size, Rate-Optimal Streaming Codes for Channels With Burst and Random Erasures," 2019, arXiv:1903.06210.
[6] A. Badr, A. Khisti, W. Tan, and J. G. Apostolopoulos, "Streaming codes for channels with burst and isolated erasures," in *Proc. INFOCOM, Turin, Italy, April 14-19*, 2013, pp. 2850–2858.
[7] S. L. Fong, A. Khisti, B. Li, W. Tan, X. Zhu, and J. G. Apostolopoulos, "Optimal streaming codes for channels with burst and arbitrary erasures," *IEEE Trans. Inf. Theory*, vol. 65, no. 7, pp. 4274–4292, 2019.
[8] M. N. Krishnan and P. V. Kumar, "Rate-Optimal Streaming Codes for Channels with Burst and Isolated Erasures," in *Proc. Int. Symp. Inf. Theory, Vail, CO, USA, June 17-22, 2018*. IEEE, pp. 1809–1813.
[9] E. Domanovitz, S. L. Fong, and A. Khisti, "An explicit rate-optimal streaming code for channels with burst and arbitrary erasures," 2019, arXiv:1904.06212.