Design and Analysis of Systematic BATS Codes

Licheng Mao, Xuan Huang, Yanyan Dong and Shenghao Yang

***Abstract*—This paper is eligible for the Jack Keil Wolf ISIT Student Paper Award. Systematic codes take the message symbols into a part of the codeword and have some practical benefits for low-latency communications. A BATS code is an efficient random linear network coding scheme that has an outer code formed by a matrix generalized fountain code and an inner code formed by random linear coding. In this paper, we design BATS codes to obtain the benefits of systematic codes while preserving the advantage of network coding. We extend the approach of fountain codes to design a systematic outer code. The existing random linear inner codes result in the expected number of message packets in a batch decreasing exponentially fast with the network length in a network with packet loss. We introduce new inner codes that provide extra protection for the message packets during transmission and show that it is possible to significantly increase the expected number of message packets in a received batch at the destination node without harm to the coding rate. Our inner codes can be applied to batched network codes with other outer codes.**

I. Introduction

Network coding has great advantages compared with traditional coding in network communications [1]–[3]. Random linear network coding (RLNC) provides a decentralized approach for network coding and achieves the multicast capacity of networks with packet loss in a broad setting [4]–[7]. Batched network codes are a class of efficient RLNC schemes for multihop network communications [8]–[13]. A batched network code comprises an outer code and an inner code. The outer code encodes the data into a sequence of batches, each of which is a number of coded symbols, and the inner code is formed by random linear coding as used in RLNC.

In traditional coding theory, a code is said to be *systematic* if all message symbols form a subset of the coded symbols [14]. Many practical codes can be designed to be systematic, for example, Reed-Solomon codes [15], fountain codes [16] and polar codes [17]. Systematic codes can benefit low-delay communications [18]–[20]. In particular, systematic RLNC schemes have been extensively studied for latency sensitive applications [21]–[27].

The benefits of the systematic codes are also attractive for batched network codes. As far as we known, however, no systematic batched network codes have been discussed in literature. In this paper, we study the systematic design of BATS codes, which are batched network codes with a matrix generalized fountain code as the outer code (see Section II for a brief, self-contained introduction). We design BATS codes including the outer code and the inner code to obtain the benefits of systematic codes while preserving the advantages of network coding. Though we focus on BATS codes in this paper, our inner code design can be applied to other batched network codes.

When the batch size is 1, the outer code becomes a fountain code and has a systematic design. When the batch size is larger than 1, the degree distribution of the outer code depends on the rank distribution of the batch transfer matrices and hence is not universal. We extend the approach of fountain codes to design a systematic outer code (see Section III). For the systematic outer code, a number of *systematic batches* are firstly generated, which consist of a partition of the message packets. Among the systematic batches, the total number of non-message packets is 0 for many cases and is no more than the batch size for all the cases in our numerical evaluations. Our systematic outer code employs the incremental encoding style [28], which can handle the uncertainty of the rank distribution of the batch transfer matrices.

To obtain the benefits of a systematic outer code, the inner code must be considered as well. For the random linear inner code, the destination node receives almost no message packets, even without packet loss. For the systematic inner codes [29], if the packet loss rate for each communication link is bounded below by a positive number, the number of message packets that can be received by the destination node decreases exponentially fast as the network length increases. To prevent the significant decrease of the number of message packets, recovery of message packets at intermediate nodes becomes necessary. Note that the recovery of message packets of a batch is not the same as decoding a batch. As the recoded packets generated by linear combinations are dense in BATS codes, the "on-the-fly decoding" employed for sparse RLNC in [27] is not optimal for recovering message packets in a batch.

In Section IV, we give a necessary and sufficient condition such that a message packet in a batch is recoverable, and show that using Gauss-Jordan elimination can find all the recoverable message packets in a batch. We also analyze the recovery of the message packets at the next hop subject to packet loss and side information. Our analysis shows that generating all recoded packets using random linear coding is not preferred and knowing more information about recoding than the coefficient vectors does not help for recovering message packets.

Based on our analysis, we improve systematic inner codes to provide extra protection of the message packets in a batch. As Gauss-Jordan elimination is only performed within a batch,

L. Mao, Y. Dong and S. Yang are with the School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Shenzhen, China. X. Huang is with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong, China.

it does not increase the computational complexity for the inner code. Our inner codes can achieve the same coding rate as the existing inner codes, meanwhile, significantly improving the number of received message packets. By tuning a parameter, the number of received message packets can be further increased using our inner code with the cost of lower coding rates. *Omitted proofs can be found in Appendix.*

II. BATS Code Basics

We briefly introduce the existing BATS codes applied on a line network. Readers are referred to [29] for a detailed discussion. A line network of length L is formed by a sequence of network nodes labeled by $0, 1, \ldots, L$, where the first node 0 is the source node and the last node L is the destination node. All the other nodes are called intermediate nodes. Network links exist only between two consecutive network nodes, modeled by packet erasure channels, i.e., a packet transmitted on a network link is either correctly received or erased.

The finite field of size q, denoted as $\mathbb{F}_q$, is called the base field. A packet of length T is a column vector in $\mathbb{F}_q^T$, and a set of packets of the same length is equated to the matrix formed by juxtaposing the packets in the set. We consider the transmissions of K message packets which form the $T \times K$ matrix $\mathbf{B}$ from the source node to the destination node. A BATS code is formed by an outer code and an inner code.

A. Ordinary Outer Code

The outer code encodes the K message packets in two steps. The first step uses a systematic code to generate a number of redundant packets, which is also called *precoding*. Let $K' \geq K$ be the total number of packets containing message packets and precoded packets. The generator matrix of the precode is of the form $[\mathbf{I}_K \ \mathbf{P}]$, where $\mathbf{I}_k$ is the $K \times K$ identity matrix and $\mathbf{P}$ is a $K \times (K' - K)$ matrix. The precoded packets are $\mathbf{B}' = [\mathbf{B} \ \mathbf{BP}]$. For our discussion, we only need the property that the precode can recover the K message packets from an η fraction of the precoded packets.

The second step encodes the precoded packets $\mathbf{B}'$ into batches of coded packets. Let M be a positive integer called the *batch size*, which is usually less than a hundred. For $i = 1, 2, \ldots$, the ith batch $\mathbf{X}_i$ includes M packets generated from the packets in $\mathbf{B}'$ using the following steps:

1) Sample a *degree distribution* $\mathbf{\Psi} = (\Psi_1, \ldots, \Psi_{\lfloor \frac{M}{1-\eta} \rfloor})$ which returns a *degree* d_i with probability Ψ_i.
2) Uniformly at random choose d_i precoded packets from $\mathbf{B}'$ and juxtaposing them into a matrix $\mathbf{B}'_i$.
3) The ith batch of $\mathbf{X}_i$ is generated by $\mathbf{X}_i = \mathbf{B}'_i\mathbf{G}_i$, where $\mathbf{G}_i$ is a $d_i \times M$ uniformly random matrix called the *batch generator matrix*.

The random values in the encoding process can be known for decoding by sharing the same pseudorandom number generator at the source node and the destination node. For a given number n of batches to be generated, the *rate of the outer code* is defined as K/n (packets per batch).

This outer code is called an *ordinary* outer code, in contrast to the systematic outer code to be discussed. For a batch of degree d, the probability that a coded packet is equal to a precoded packet is dq^{-d}. As not all precoded packets are message packets, the probability that a coded packet is equal to a message packet is no greater than dq^{-d}. Typically, $d \geq M \geq 4$ and $q = 256$. So, it is unlikely that a message packet appears in a batch using the ordinary outer code.

B. General Inner Code

The inner code is the composition of the *recoding* operations performed on each batch separately. At the source node, the packets generated by recoding for a batch are the linear combinations of the M packets generated by the outer code. At an intermediate node, the packets generated by recoding for a batch are the linear combinations of packets received by the node belonging to the batch.

Fix a certain network node v. Let $\mathbf{Y}_i^v$ be the received packets of the ith batch at the node v. At the source node $\mathbf{Y}_i^0 = \mathbf{X}_i$. As recoding is linear, for $v = 1, \ldots, L$,

$$\mathbf{Y}_i^v = \mathbf{X}_i\mathbf{H}_i^v = \mathbf{B}'_i\mathbf{G}_i\mathbf{H}_i^v, \tag{1}$$

where $\mathbf{H}_i^v$ is called the *(batch) transfer matrix* of the ith batch at the node v. The number of rows of $\mathbf{H}_i^v$ is M. The number of columns of $\mathbf{H}_i^v$ corresponds to the number of packets received for the ith batch at the node v, which may vary for different batches and is finite. If no packets are received for a batch, $\mathbf{Y}_i^v$ ($\mathbf{H}_i^v$) is the empty matrix of 0 columns.

Suppose that *coefficient vectors* are embedded in the packet header right after $\mathbf{X}_i$ is generated, where the matrix formed by the coefficient vectors is the identity matrix. The same linear operations performed on the batch are performed on the coefficient vectors as well [4]. By this approach, $\mathbf{H}_i^v$ can be known at each node v that receives batch i from the header of the batch.

We say a set of packets of a batch are linearly independent/dependent if their corresponding coefficient vectors in the packet header are linearly independent/dependent. We call $\mathrm{rank}(\mathbf{H}_i^v)$ the *rank of the ith batch* at node v. Assume that the transfer matrices $\mathbf{H}_i^v$ have the same *rank distribution* $\mathbf{h}^v = (h_0^v, h_1^v, \ldots, h_M^v)$ at the node v.

C. Decoding Algorithms

The outer code can be regarded as a channel code for the matrix multiplication channel $\mathbf{Y} = \mathbf{X}\mathbf{H}^L$ where $\mathbf{H}^L$ has the rank distribution $\mathbf{h}^L$ and is known by the receiver. The achievable rate of the outer code is upper bounded by the capacity of this channel, which is the expected rank $\mathbb{E}[\mathbf{h}^L] = \sum_{i=1}^{M} i h_i^L$ (packets per batch) [30].

The destination node can decode the received batches in two steps: The first step recovers a fraction of precoded packets using a belief propagation algorithm [29]. The second step decodes the precoded packets to recover the message packets. According to the existing theory of BATS codes [29], it is possible to design a degree distribution $\mathbf{\Psi}$ for a given rank distribution $\mathbf{h}^L$ such that when K is large, the BP decoding can recover a given fraction of the precoded packets, and the achievable rate of the BP decoding is very close to

$\mathbb{E}[\mathbf{h}^L]$. Specifically, we only need slightly more than $K/\mathbb{E}[\mathbf{h}^L]$ batches to recover the K message packets.

When K is relatively small, BP decoding tends to stop before decoding a large fraction of the input packets. Though we can continue decoding by Gaussian elimination, the computational complexity is high. A better approach is to use *inactivation decoding*: when BP decoding stops, an undecoded message packet is marked as inactive and substituted into the batches as a decoded packet to resume the BP decoding procedure. Inactivation decoding reduces the complexity of Gaussian elimination and improves the success probability of BP decoding.

III. Systematic Outer Codes

We study how to design an outer code such that for some batches, all or most coded packets are message packets and all the message packets are included in these batches. Such batches are called *systematic batches*, and such an outer code is said to be *systematic*.

A. Problem Formulation

Different from fountain codes, the degree distribution of the outer code of a BATS code depends on the rank distribution of the batch transfer matrices at the destination node. Therefore, the systematic outer code design also depends on the rank distribution. We study the systematic outer code for a composite scenario that consists of two cases of the rank distribution:

- Case I: the network links have no loss and all the batch transfer matrices have rank M. The rank distribution in this case is said to be *full-rank*;
- Case II: the network links are lossy and the rank distribution may not be full-rank.

For Case I, only the systematic batches are required to be transmitted, and all the message packets can be received by the destination node directly without decoding. For Case II, in addition to the systematic batches, more batches generated by the ordinary outer code (called *ordinary batches*) should be transmitted. The message packets can be decoded at the destination node when a sufficient number of (systematic and ordinary) batches is received.

The source node does not need to know which case occurs beforehand: it first transmits the systematic batches, followed by the ordinary batches, until it receives feedback notifying the correct decoding at the destination node or a practical limit on the total number of batches to transmit is reached. This kind of outer code is also called *incremental encoding*, which has been discussed without the systematic outer code [28].

Suppose we have K message packets $\mathbf{B}$ for encoding using a systematic outer code with batch size M. Let n be an integer larger than or equal to K/M to be decided later. We want to design an outer code such that the first n batches are systematic batches that include all the message packets. When $M = 1$, the outer code becomes a Raptor code, and a systematic Raptor code can be designed with $n = K$ [31]. Our approach for $M > 1$ generalized the systematic Raptor codes where nM can be larger than K.

TABLE I

Here $M = 32$. For each value of $K = 5M, 25M, 50M, 5000$ instances of the ordinary outer code are sampled. The table gives the number of consistent instances when $n = K/M, K/M + 1, K/M + 2$ and $n \geq K/M + 3$.

$n - K/M$	$K = 5M$	$K = 25M$	$K = 50M$
0	999	3	0
1	1200	1110	997
2	694	373	195
≥ 3	2107	3514	3808

B. Deterministic Encoder and Decoder

We introduce a pair of deterministic encoder and decoder for the ordinary outer code that will be used in our systematic outer code design. Let $\mathbf{\Psi}^M$ be the degree distribution optimized for the full-rank rank distribution as in [29, Chapter 6], which achieves the optimal rate of the outer code asymptotically when the batch transfer matrices are the identity matrices.

The ordinary outer code is a random code, and an instance of the ordinary outer code is determined by the degree, the set of precoded packets sampled and the batch generator matrix for each batch. Denote by ENC_n an instance of the ordinary outer code encoder with the degree distribution $\mathbf{\Psi}^M$ for generating n batches. For K message packets $\mathbf{B}$, $(\mathbf{X}_1, \ldots, \mathbf{X}_n) = \mathrm{ENC}_n(\mathbf{B})$ are the n batches generated. Denote by DEC_n the corresponding decoder of ENC_n. We say such a pair $(\mathrm{ENC}_n, \mathrm{DEC}_n)$ is *consistent* if for any K packets $\mathbf{B}$

$$\mathbf{B} = \mathrm{DEC}_n\big(\mathrm{ENC}_n(\mathbf{B})\big),$$

where the batches generated by ENC_n are decoded directly, so that the batch transfer matrices are the identity matrix.

As the degree distribution $\mathbf{\Psi}^M$ is optimized for the full-rank rank distribution, the ordinary outer code can guarantee a high decoding successful probability when n is sufficiently larger than K/M [29]. For a given n no less than K/M, we can perform random trials of the ordinary outer code until a consistent instance $(\mathrm{ENC}_n, \mathrm{DEC}_n)$ is found. It is not necessary that all or most the instances are consistent, as our purpose is to find only one consistent instance.

We performed some experiments with results in Table I. We observe that when K is relatively small, consistent instance with $n = K/M$ can be found, and when K is relatively large, there are still a significant fraction of consistent instances with $n = K/M + 1$.

C. Systematic Outer Code

Fix a consistent pair $(\mathrm{ENC}_n, \mathrm{DEC}_n)$. The decoder DEC_n solves K message packets from the nM coded packets. Among the nM coded packets, $nM - K$ coded packets are redundant, which will not be used by the decoder. All the redundant packets can be identified by a trial of DEC_n (e.g., on the all-zero dummy coded packets). Let M_i be the number of non-redundant coded packets in the ith batch. Denote by DEC_n^* the same decoder as DEC_n except that the redundant input coded packets are removed from the decoder input.

The systematic outer code works as follows:

1) Partition the message packets $\mathbf{B}$ into n subsets, where the number of packets in the ith subset $\tilde{\mathbf{X}}_i$ is M_i.
2) Calculate $\tilde{\mathbf{B}} = \mathrm{DEC}_n^*(\tilde{\mathbf{X}}_1, \ldots, \tilde{\mathbf{X}}_n)$.
3) Generate the n systematic batches $\mathrm{ENC}_n(\tilde{\mathbf{B}})$.
4) Generate more (ordinary) batches by performing the ordinary outer code on $\tilde{\mathbf{B}}$.

Recall that both the systematic batches and the ordinary batches are used for the decoding in Case II. See [28] for an approach to design the degree distribution used in the ordinary outer code in Step 4) to optimize the decoding performance.

IV. Inner Code for Systematic Batches

In a systematic batch, all or some of the coded packets are message packets. However, the inner code to be further applied to the batch and the packet erasure during the transmission can potentially reduce the number of message packets in the batch. In this section, we analyze how the inner code and packet erasure affect the message packets in systematic batches, and study inner codes that can protect the message packets in a systematic batch.

A. General Inner Code Formulation

We consider the inner code on a line network as described in Section II. As the inner code is performed on each batch individually, we consider a generic batch $\mathbf{X}$ without the subscripts. We call the packets in batch $\mathbf{X}$ the *x-packets* for convenience. By (1), the received packets $\mathbf{Y}^u$ of the batch $\mathbf{X}$ at node u satisfies

$$\mathbf{Y}^u = \mathbf{X}\mathbf{H}^u, \tag{2}$$

where $\mathbf{H}^u$ is the transfer matrix of the batch at the node u.

Let N_u be the number of columns of $\mathbf{Y}^u$ (or $\mathbf{H}^u$), i.e., the number of received packets of the batch at node u. For a non-destination node u, we use $u+$ to denote the receiver of the outgoing link of u in the line network. Suppose that the node u needs to transmit N'_u packets of the batch $\mathbf{X}$ to the node $u+$. The transmitted packets, called *recoded packets*, are generated by linear combinations as $\mathbf{Y}^u\mathbf{\Phi}^u = \mathbf{X}\mathbf{H}^u\mathbf{\Phi}^u$, where $\mathbf{\Phi}^u$ is an $N_u \times N'_u$ matrix over the base field. Due to packet loss, the set of received packets at $u+$ is a subset of $\mathbf{Y}^u\mathbf{\Phi}^u$. Let $\mathbf{E}^u$ be an $N'_u \times N_{u+}$ matrix obtained by removing columns of identity matrix specifying the packet erasures. We can write

$$\mathbf{Y}^{u+} = \mathbf{X}\mathbf{H}^u\mathbf{\Phi}^u\mathbf{E}^u = \mathbf{X}\mathbf{H}^{u+}, \tag{3}$$

where $\mathbf{H}^{u+} = \mathbf{H}^u\mathbf{\Phi}^u\mathbf{E}^u$.

We first review two existing inner code schemes.

Random Linear Inner Code: All the recoded packets are generated by *uniformly random linear combinations*, i.e., $\mathbf{\Phi}^u$ is a uniformly random matrix over the base field. At the source node, the probability that a recoded packet (a column of $\mathbf{X}\mathbf{\Phi}^u$) is an x-packet is q^{-M}.

Systematic Inner Code: The systematic inner code uses all the linearly independent received packets as recoded packets and generates other recoded packets by uniformly random linear combinations. If each packet is transmitted with erasure probability $\epsilon > 0$ independently at each link, the number of received x-packets at the destination node drops exponentially fast with L increasing.

B. Recovery of x-Packets

For the existing inner code schemes discussed above, the received packets $\mathbf{Y}^u$ at a non-source node u may include no or only a small number of x-packets. If $\mathrm{rank}(\mathbf{H}^u) = M$, however, all the x-packets can be recovered by solving the system (2). So we are motivated to study whether some of the x-packets can be recovered in general by operations within a batch.

We say *the ith x-packet can be recovered at node u* if the system (2) has a unique solution for the ith column of $\mathbf{X}$. Denote by $\mathrm{Col}(\mathbf{H}^u)$ the column space of a matrix $\mathbf{H}^u$. Let $\mathbf{e}_i$ be a length-M column vector with its ith entry 1 and all other entries 0.

Lemma 1. *Under the condition that $\mathbf{Y}^u = \mathbf{X}\mathbf{H}^u$ has at least one solution, the ith x-packet can be recovered if and only if $\mathbf{e}_i \in \mathrm{Col}(\mathbf{H}^u)$.*

The following proposition tells us that we can test the recoverability of all x-packets from the reduced column echelon form of $\mathbf{H}^u$, which can be obtained by column-wise Gauss-Jordan elimination.

Proposition 2. *Let $\mathbf{L}$ be the reduced column echelon form for a matrix $\mathbf{H}^u$. Then $\mathbf{e}_i \in \mathrm{Col}(\mathbf{H}^u)$ if and only if $\mathbf{e}_i$ is a column of $\mathbf{L}$.*

Let $r = \mathrm{rank}(\mathbf{H}^u)$ and $\mathbf{V}$ be an $N_u \times N_u$ matrix such that $\mathbf{H}^u\mathbf{V}$ is in reduced column echelon form. To recover x-packets, we perform the same column operations on $\mathbf{Y}^u$ and obtain $\mathbf{Y}^u\mathbf{V} = \mathbf{X}\mathbf{H}^u\mathbf{V}$. If $\mathbf{e}_i$ is a column of $\mathbf{H}^u\mathbf{V}$, then that column of $\mathbf{Y}^u\mathbf{V}$ equals to the ith x-packet.

Let s be the number of x-packets that can be recovered from $\mathbf{Y}^u$. By Proposition 2, there are exactly s distinct columns in $\mathbf{H}^u\mathbf{V}$ with only 1 non-zero entry being one. Therefore, by proper row and column permutations, $\mathbf{H}^u\mathbf{V}$ is of the form

$$\begin{bmatrix} \mathbf{I}_s & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{r-s} & \mathbf{0} \\ \mathbf{0} & \mathbf{T} & \mathbf{0} \end{bmatrix}, \tag{4}$$

where $\mathbf{I}_k$ is the $k \times k$ identity matrix, $\mathbf{0}$ is an all-zero matrix of proper size, and $\mathbf{T}$ is an $(N_u - r) \times r$ matrix where each column is not zero. Denoting the first r columns of the corresponding column permutation matrix as the $N_u \times r$ matrix $\mathbf{P}$, exactly each of the first s columns of $\mathbf{H}^u\mathbf{V}\mathbf{P}$ has only 1 non-zero entry.

We discuss some general properties about the recovery of x-packets at the node $u+$, which provide guidance on designing new inner codes.

Proposition 3. *If an x-packet cannot be recovered at the node u, then it cannot be recovered at the node $u+$ on the next hop.*

In the following two claims, we assume that $\mathbf{X}$, $\mathbf{H}^u$, $\mathbf{\Phi}^u$ and $\mathbf{E}^u$ as described in (3) are mutually independent. They

show that knowing more information about the inner code at node u (e.g., $\mathbf{\Phi}^u$) than $\mathbf{H}^{u+}$ and $\mathbf{Y}^{u+}$ at the node $u+$ does not help for recovering x-packets at the node $u+$.

Lemma 4. *$\mathbf{\Phi}^u$ and $\mathbf{X}$ are conditionally independent given $\mathbf{H}^{u+}$ and $\mathbf{Y}^{u+}$.*

Proposition 5. *Let S be any random variable that is conditionally independent with $\mathbf{X}$ given $\mathbf{H}^{u+}$ and $\mathbf{Y}^{u+}$. Given the instance of $\mathbf{H}^{u+}$ and $\mathbf{Y}^{u+}$ at the node $u+$, further knowing the instance of S at the node $u+$ does not help to recover more x-packets at $u+$.*

The following proposition states that using the random linear inner code at node u, the node $u+$ can recover almost no x-packets when the number of received packets at $u+$ is fewer than $\mathrm{rank}(\mathbf{H}^u)$. Denote by $\zeta_k^{m,n}$ the probability that an $m \times n$ uniformly random matrix over $\mathbb{F}_q$ has rank k. See, e.g., [29, Sec. 3.3.2] for a formula of $\zeta_k^{m,n}$.

Proposition 6. *Suppose the random linear inner code over $\mathbb{F}_q$ is used at the node u and $N_{u+} < r = \mathrm{rank}(\mathbf{H}^u)$. Under the condition that $\mathbf{e}_i \in \mathrm{Col}(\mathbf{H}^u)$, the probability that $\mathbf{e}_i \in \mathrm{Col}(\mathbf{H}^{u+})$ is $1 - \sum_{k=0}^{N_{u+}} \zeta_k^{r-1,N_{u+}} q^{k-N_{u+}}$ and it converges to zero as $q \to \infty$.*

It is unavoidable that the number of received packets at $u+$ is fewer than $\mathrm{rank}(\mathbf{H}^u)$ due to packet loss. Together with Proposition 3, Proposition 6 implies that as long as the event $N_{u+} < \mathrm{rank}(\mathbf{H}^u)$ occurs once at some node u, the destination node receives almost no message packets from a systematic batch. Therefore, random linear recoding is not preferred for recovering x-packets. So, we are motivated to extend systematic inner codes for the recovery of x-packets.

C. Inner Codes with x-Packet Protection

We propose two inner codes that can protect the x-packets during recoding. We first define two recoding matrices. Suppose that s x-packets are recoverable at the node u.

1) x-Protection Recoding: For an integer w with $0 \le w \le N'_u - s$, let $\mathbf{R}_w$ be an $r \times N'_u$ matrix of the form

$$\mathbf{R}_w = \begin{bmatrix} \mathbf{I}_s & \mathbf{U}_{s,w} & \mathbf{U}_{r,N'_u-s-w} \\ \mathbf{0} & \mathbf{0} & \end{bmatrix},$$

where $\mathbf{U}_{m,n}$ is the $m \times n$ uniformly random matrix.

2) Systematic x-Protection Recoding: For an integer w with $0 \le w \le N'_u - s$, let $\mathbf{R}^{\mathrm{s}}_w$ be an $r \times N'_u$ matrix of the form: when $w < N'_u - r$,

$$\mathbf{R}^{\mathrm{s}}_w = \begin{bmatrix} \mathbf{I}_s & \mathbf{U}_{s,w} & \mathbf{0} & \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{r-s} & \mathbf{U}_{r,N'_u-r-w} \end{bmatrix};$$

when $w \ge N'_u - r$,

$$\mathbf{R}^{\mathrm{s}}_w = \begin{bmatrix} \mathbf{I}_s & \mathbf{U}_{s,w} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{J} \end{bmatrix},$$

where $\mathbf{J}$ is the first $N'_u - w - s$ columns of $\mathbf{I}_{r-s}$.

The inner code operations at node u consist of i) the Gauss-Jordan elimination represented by the matrix $\mathbf{V}$, ii) the column permutation and removing the all-zero columns represented by the matrix $\mathbf{P}$, and iii) (systematic) x-protection recoding $\mathbf{R}_w$ ($\mathbf{R}^{\mathrm{s}}_w$). When the overall recoding matrix at node u is $\mathbf{VPR}_w$, the inner code is call the *x-protection inner code (XPIC)*. When the overall recoding matrix at node u is $\mathbf{VPR}^{\mathrm{s}}_w$, the inner code is call the *systematic x-protection inner code (SXPIC)*.

The value of w controls the level of protection of x-packets. When $w = 0$, no extra protection is provided for x-packets, and we can check that SXPIC has the same rank performance of the systematic inner code. When $w = N'_u - s$, all recoded packets generated by linear combinations are used for protecting the x-packets.

We perform numerical evaluations to verify the performance of the new inner codes in terms of both the average rank and the average number of recoverable x-packets, and compare it with those of the random linear inner code (RLIC) and the systematic inner code (SIC). We use the line networks of length up to 50 hops, where each link has the same independent packet erasure probability 0.2. The batch size $M = 16$ and the number of packets to transmit $N'_u = 20$ for all node u. Since the performance of SXPIC and XPIC have negligible differences in simulation, we only show the results for SXPIC, where we evaluate $w = 0$ and $w = N'_u - s$ as representative.

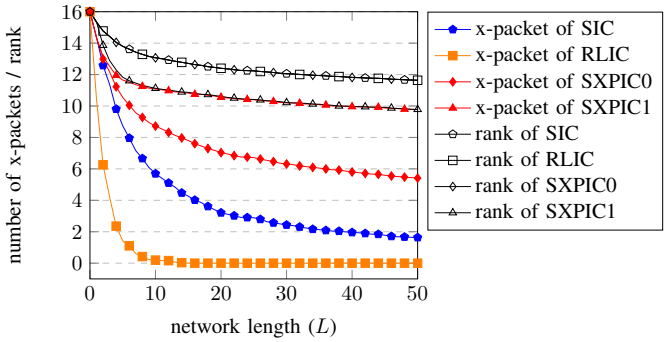

Fig. 1. The average number of recoverable x-packets and the average rank at node 0 to 50 for SIC, RLIC, SXPIC with $w = 0$ (denoted by SXPIC0) and SXPIC with $w = N'_u - s$ (denoted by SXPIC1), each with 500 trials.

Our numerical evaluation results are shown in Figure 1. We plot the average number of recoverable x-packets and the average rank at node 0 to 50 for SIC, RLIC, SXPIC with $w = 0$ (denoted by SXPIC0) and SXPIC with $w = N'_u - s$ (denoted by SXPIC1), each with 500 trials. We see that for SIC and RLIC, the average number of recoverable x-packets drops fast. SXPIC0 has a much higher average number of recoverable x-packets than that of SIC and RLIC, while preserving almost the same average rank. On the other hand, SXPIC1 has the highest average number of recoverable x-packets among the four inner codes, at a cost of reduced average rank.

V. Concluding Remarks

We proposed the first design of batched network codes that can benefit from the advantages of systematic codes. This kind of systematic batched network codes can find applications in latency sensitive applications in network communications.

REFERENCES

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[2] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inform. Theory*, vol. 49, no. 2, pp. 371–381, Feb. 2003.

[3] R. Koetter and M. Medard, "An algebraic approach to network coding," *IEEE/ACM Trans. Networking*, vol. 11, no. 5, pp. 782–795, Oct. 2003.

[4] T. Ho, B. Leong, M. Medard, R. Koetter, Y. Chang, and M. Effros, "The benefits of coding over routing in a randomized setting," in *Proc. IEEE ISIT '03*, Jun. 2003.

[5] 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 Trans. Inform. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.

[6] S. Jaggi, P. A. Chou, and K. Jain, "Low complexity optimal algebraic multicast codes," in *Proc. IEEE ISIT '03*, Jun. 2003.

[7] P. Sanders, S. Egner, and L. Tolhuizen, "Polynomial time algorithms for network information flow," in *Proc. ACM SPAA '03*, New York, NY, USA, 2003, pp. 286–294.

[8] D. Silva, W. Zeng, and F. R. Kschischang, "Sparse network coding with overlapping classes," in *Proc. NetCod '09*, Jun. 2009, pp. 74–79.

[9] A. Heidarzadeh and A. H. Banihashemi, "Overlapped chunked network coding," in *Proc. ITW '10*, Jan. 2010, pp. 1–5.

[10] Y. Li, E. Soljanin, and P. Spasojevic, "Effects of the generation size and overlap on throughput and complexity in randomized linear network coding," *IEEE Trans. Inform. Theory*, vol. 57, no. 2, pp. 1111–1123, Feb. 2011.

[11] S. Yang and R. W. Yeung, "Batched sparse codes," *IEEE Trans. Inform. Theory*, vol. 60, no. 9, pp. 5322–5346, Sep. 2014.

[12] B. Tang, S. Yang, Y. Yin, B. Ye, and S. Lu, "Expander chunked codes," *EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, pp. 1–13, Dec. 2015.

[13] B. Tang and S. Yang, "An LDPC approach for chunked network codes," *IEEE/ACM Trans. Networking*, vol. 26, no. 1, pp. 605–617, Feb. 2018.

[14] F. MacWilliams and N. Sloane, *The Theory of Error-Correcting Codes*. North-Holland publishing, 1978.

[15] D. J. Versfeld, J. N. Ridley, H. C. Ferreira, and A. S. Helberg, "On systematic generator matrices for reed–solomon codes," *IEEE Trans. Inform. Theory*, vol. 56, no. 6, pp. 2549–2550, 2010.

[16] M. Luby, A. Shokrollahi, M. Watson, T. Stockhammer, and L. Minder, "RaptorQ forward error correction scheme for object delivery – rfc 6330," 2011. [Online]. Available: http://datatracker.ietf.org/doc/rfc6330/

[17] E. Arikan, "Systematic polar coding," *IEEE communications letters*, vol. 15, no. 8, pp. 860–862, 2011.

[18] A. Badr, A. Khisti, W.-T. Tan, and J. Apostolopoulos, "Perfecting protection for interactive multimedia: A survey of forward error correction for low-delay interactive applications," *IEEE Signal Processing Magazine*, vol. 34, no. 2, pp. 95–113, 2017.

[19] A. Garcia-Saavedra, M. Karzand, and D. J. Leith, "Low delay random linear coding and scheduling over multiple interfaces," *IEEE Transactions on Mobile Computing*, vol. 16, no. 11, pp. 3100–3114, 2017.

[20] Y. Li, F. Zhang, J. Wang, T. Q. S. Quek, and J. Wang, "On streaming coding for low-latency packet transmissions over highly lossy links," *IEEE Communications Letters*, vol. 24, no. 9, pp. 1885–1889, 2020.

[21] R. Prior and A. Rodrigues, "Systematic network coding for packet loss concealment in broadcast distribution," in *The International Conference on Information Networking 2011 (ICOIN2011)*, 2011, pp. 245–250.

[22] D. E. Lucani, M. Medard, and M. Stojanovic, "On coding for delay—network coding for time-division duplexing," *IEEE Trans. Inform. Theory*, vol. 58, no. 4, pp. 2330–2348, 2012.

[23] M. Yu, N. Aboutorab, and P. Sadeghi, "From instantly decodable to random linear network coded broadcast," *IEEE Transactions on Communications*, vol. 62, no. 11, pp. 3943–3955, 2014.

[24] F. Gabriel, S. Wunderlich, S. Pandi, F. H. P. Fitzek, and M. Reisslein, "Caterpillar rlnc with feedback (crlnc-fb): Reducing delay in selective repeat arq through coding," *IEEE Access*, vol. 6, pp. 44 787–44 802, 2018.

[25] C. V. Phung, A. Engelmann, and A. Jukan, "Error correction with systematic rlnc in multi-channel thz communication systems," in *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2020, pp. 512–517.

[26] F. Karetsi and E. Papapetrou, "A low complexity network-coded arq protocol for ultra-reliable low latency communication," in *2021 IEEE 22nd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2021, pp. 11–20.

[27] E. Tasdemir, M. Tömösközi, J. A. Cabrera, F. Gabriel, D. You, F. H. P. Fitzek, and M. Reisslein, "Sparec: Sparse systematic rlnc recoding in multi-hop networks," *IEEE Access*, vol. 9, pp. 168 567–168 586, 2021.

[28] X. Xu, Y. L. Guan, Y. Zeng, and C. C. Chui, "Quasi-universal BATS code," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 3497–3501, April 2017.

[29] S. Yang and R. W. Yeung, *BATS Codes: Theory and Practice*, ser. Synthesis Lectures on Communication Networks. Morgan & Claypool Publishers, 2017.

[30] S. Yang, J. Meng, and E.-h. Yang, "Coding for linear operator channels over finite fields," in *Proc. IEEE ISIT '10*, Jun. 2010, pp. 2413–2417.

[31] A. Shokrollahi and M. Luby, *Raptor Codes*, ser. Foundations and Trends in Communications and Information Theory. now, 2011, vol. 6.

APPENDIX A
PROOF OF RECOVERABILITY OF X-PACKETS

A. Proof of Lemma 1

Proof: For convenience, we omit the superscripts of $\mathbf{Y}^u$ and $\mathbf{H}^u$ in this proof. Let $\mathbf{X}^T = [\bar{\mathbf{x}}_1, \ldots, \bar{\mathbf{x}}_T]$ and $\mathbf{Y}^T = [\bar{\mathbf{y}}_1 \ldots, \bar{\mathbf{y}}_T]$, in which $\bar{\mathbf{x}}_j$, $\bar{\mathbf{y}}_j, j = 1, \ldots, T$ are column vectors. Solving the equation $\mathbf{Y} = \mathbf{X}\mathbf{H}$ is equivalent with solving $\mathbf{Y}^T = \mathbf{H}^T\mathbf{X}^T$, or

$$\mathbf{H}^T\bar{\mathbf{x}}_j = \bar{\mathbf{y}}_j, j = 1, \ldots, T.$$

Since there is at least one solution, there exists a particular solution $\bar{\mathbf{x}}_j^p$, $j = 1, \ldots, T$ for this system. The solution set is

$$\bar{\mathbf{x}}_j^p + \mathrm{Null}(\mathbf{H}^T), j = 1, \ldots, T,$$

where $\mathrm{Null}(\mathbf{H}^T)$ is the the null space of $\mathbf{H}^T$. The solution for $\mathbf{x}_i$ is unique if and only if ith position of any vector in $\mathrm{Null}(\mathbf{H}^T)$ is 0, which is equivalent to $\mathbf{e}_i$ is orthogonal to $\mathrm{Null}(\mathbf{H}^T)$. As $\mathrm{Col}(\mathbf{H})$ is the orthogonal complement of $\mathrm{Null}(\mathbf{H}^T)$, the proof is completed. ∎

B. Proof of Proposition 2

Proof: For the sufficiency, if $\mathbf{e}_i$ is a column of $\mathbf{L}$, $\mathbf{e}_i \in \mathrm{Col}(\mathbf{L}) = \mathrm{Col}(\mathbf{H}^u)$.

Now we prove the necessity. If $\mathbf{e}_i \in \mathrm{Col}(\mathbf{H}^u)$, then $\mathbf{e}_i \in \mathrm{Col}(\mathbf{L})$. Suppose $\mathbf{e}_i = \sum_j c_j\mathbf{l}_j$, where c_j are constants and $\mathbf{l}_j$ is jth non-zero column of $\mathbf{L}$. By the property that all of the zero columns are in the right of the non-zero columns in $\mathbf{L}$, jth non-zero column of $\mathbf{L}$ is jth column of $\mathbf{L}$. Denote by r_j the row index of the leading coefficient 1 of $\mathbf{l}_j$, which must exist due to the property of reduced column echelon form. Further, as the (r_j, j) entry is the only non-zero entry on the r_jth row of $\mathbf{L}$, we have $c_j = 0$ for all j such that $r_j \neq i$ and $c_j = 1$ for j such that $r_j = i$. Thus $\mathbf{e}_i = \mathbf{l}_{j^*}$ with $r_{j^*} = i$. The proof is completed. ∎

C. Proof of Proposition 3

Proof: If the packet $\mathbf{x}_i$ cannot be recovered at the node u, by Lemma 1, we have $\mathbf{e}_i \notin \mathrm{Col}(\mathbf{H}^u)$. Due to $\mathrm{Col}(\mathbf{H}^{u+}) = \mathrm{Col}(\mathbf{H}^u\mathbf{\Phi}^u\mathbf{E}^u) \subseteq \mathrm{Col}(\mathbf{H}^u)$, $\mathbf{e}_i \notin \mathrm{Col}(\mathbf{H}^{u+})$ and thus $\mathbf{x}_i$ cannot be recovered at the node u^+. ∎

D. Proof of Lemma 4

Proof: Denote by $\mathbf{x}$, $\mathbf{y}^+$, $\mathbf{h}^+$ and ϕ the instances of $\mathbf{X}$, $\mathbf{Y}^{u+}$, $\mathbf{H}^{u+}$ and $\mathbf{\Phi}^u$, respectively. As $P(\mathbf{x}, \phi|\mathbf{y}^+, \mathbf{h}^+) = P(\phi|\mathbf{y}^+, \mathbf{h}^+)P(\mathbf{x}|\mathbf{y}^+, \mathbf{h}^+, \phi)$, to prove the claim, it is sufficient to show that $P(\mathbf{x}|\mathbf{y}^+, \mathbf{h}^+, \phi) = P(\mathbf{x}|\mathbf{y}^+, \mathbf{h}^+)$ for all instances. If $\mathbf{y}^+ \neq \mathbf{x}\mathbf{h}^+$, the equality holds as both sides are 0. Suppose $\mathbf{y}^+ = \mathbf{x}\mathbf{h}^+$. As $\mathbf{X}, \mathbf{H}^u, \mathbf{\Phi}^u$ and $\mathbf{E}^u$ are independent and $\mathbf{H}^{u+} = \mathbf{H}^u\mathbf{\Phi}^u\mathbf{E}^u$, we get

$$\begin{aligned} P(\mathbf{x}|\mathbf{y}^+, \mathbf{h}^+, \phi) &= \frac{P(\mathbf{x})P(\mathbf{h}^+, \phi)}{P(\mathbf{y}^+, \mathbf{h}^+, \phi)} \\ &= \frac{P(\mathbf{x})P(\mathbf{h}^+, \phi)}{\sum_{\mathbf{x}':\mathbf{x}'\mathbf{h}^+=\mathbf{y}^+} p(\mathbf{x}')P(\mathbf{h}^+, \phi)} \\ &= \frac{P(\mathbf{x})}{\sum_{\mathbf{x}':\mathbf{x}'\mathbf{h}^+=\mathbf{y}^+} p(\mathbf{x}')}. \end{aligned}$$

Similarly,

$$\begin{aligned} P(\mathbf{x}|\mathbf{y}^+, \mathbf{h}^+) &= \frac{P(\mathbf{x})P(\mathbf{h}^+)}{P(\mathbf{y}^+, \mathbf{h}^+)} \\ &= \frac{P(\mathbf{x})P(\mathbf{h}^+)}{\sum_{\mathbf{x}':\mathbf{x}'\mathbf{h}^+=\mathbf{y}^+} p(\mathbf{x}')P(\mathbf{h}^+)} \\ &= \frac{P(\mathbf{x})}{\sum_{\mathbf{x}':\mathbf{x}'\mathbf{h}^+=\mathbf{y}^+} p(\mathbf{x}')}. \end{aligned}$$

∎

E. Proof of Proposition 5

Proof: Denote by $\mathbf{x}$, $\mathbf{y}^+$, $\mathbf{h}^+$ and s the instances of $\mathbf{X}$, $\mathbf{Y}^{u+}$, $\mathbf{H}^{u+}$ and S, respectively. It is sufficient to show that $P(\mathbf{x}|\mathbf{y}^+, \mathbf{h}^+, s) = P(\mathbf{x}|\mathbf{y}^+, \mathbf{h}^+)$ for all instances $\mathbf{x}, s, \mathbf{y}^+, \mathbf{h}^+$. If $\mathbf{y}^+ \neq \mathbf{x}\mathbf{h}^+$, the equality holds as both sides are 0. Suppose $\mathbf{y}^+ = \mathbf{x}\mathbf{h}^+$. As $\mathbf{X}$ and S are independent given $\mathbf{H}^{u+}$ and $\mathbf{Y}^{u+}$, we get

$$\begin{aligned} P(\mathbf{x}|\mathbf{y}^+, \mathbf{h}^+, s) &= \frac{P(\mathbf{x}, s|\mathbf{y}^+, \mathbf{h}^+)}{P(s|\mathbf{y}^+, \mathbf{h}^+)} \\ &= \frac{P(s|\mathbf{y}^+, \mathbf{h}^+)P(\mathbf{x}|\mathbf{y}^+, \mathbf{h}^+)}{P(s|\mathbf{y}^+, \mathbf{h}^+)} \\ &= P(\mathbf{x}|\mathbf{y}^+, \mathbf{h}^+). \end{aligned}$$

∎

F. Proof of Proposition 6

Proof: For convenience, we omit the subscripts of $\mathbf{H}^u$, $\mathbf{\Phi}^u$ and $\mathbf{E}^u$.

Assume that $\mathbf{H} \in \mathbb{F}_q^{M\times N_u}$ is fixed with $\mathrm{rank}(\mathbf{H}) = r$ and $\mathbf{e}_i \in \mathrm{Col}(\mathbf{H})$. Since $\mathrm{rank}(\mathbf{H}) = r$, and $\mathbf{e}_i \in \mathrm{Col}(\mathbf{H})$, we can extend $\{\mathbf{e}_i\}$ to a basis of $\mathrm{Col}(\mathbf{H})$, denoted by $\mathbf{W}$. Then there exists a full row rank matrix $\mathbf{C} \in \mathbb{F}_q^{r\times N_u}$ such that $\mathbf{H} = \mathbf{W}\mathbf{C}$ and $\mathbf{H}^{u+} = \mathbf{W}\mathbf{C}\mathbf{\Phi}\mathbf{E}$. Let $\mathbf{\Phi}^* = \mathbf{\Phi}\mathbf{E}$, then $\mathbf{\Phi}^*$ is an $N_u \times N_{u+}$ uniformly random matrix.

Notice that $\mathbf{C}$ is full row rank, $\mathbf{C}$ can be written as $\mathbf{C} = \mathbf{K}\mathbf{C}'$ where $\mathbf{C}'$ is an invertible matrix with the first r rows being $\mathbf{C}$ and $\mathbf{K}$ is made up of the first r rows of an identity matrix. Since $\mathbf{C}'\mathbf{\Phi}^*$ is still an $N_u \times N_{u+}$ uniformly random matrix, we have $\mathbf{C}\mathbf{\Phi}^*$ is an $r \times N_{u+}$ uniformly random matrix. In the following we let $\mathbf{M} = \mathbf{C}\mathbf{\Phi}^*$ and we have $\mathbf{e}_i \in \mathrm{Col}(\mathbf{H}^{u+})$ if and only if $\mathbf{e}_1 \in \mathrm{Col}(\mathbf{M})$. Let $\boldsymbol{m}^T$ be the first row of $\mathbf{M}$ and $\tilde{\mathbf{M}}$ be the submatrix of $\mathbf{M}$ with first row removed. Then $\mathbf{e}_1 \in \mathrm{Col}(\mathbf{M})$ is equivalent to $\exists\mathbf{x}$ s.t. $\tilde{\mathbf{M}}\mathbf{x} = \mathbf{0}, \boldsymbol{m}^T\mathbf{x} \neq 0$, in other words, $\boldsymbol{m} \not\perp \mathrm{Null}(\tilde{\mathbf{M}})$.

When $\tilde{\mathbf{M}}$ has rank k, the null space of $\tilde{\mathbf{M}}$ has dimension $N_{u+} - k$. The probability $\boldsymbol{m} \not\perp \mathrm{Null}(\tilde{\mathbf{M}})$ is $1 - \frac{q^k}{q^{N_{u+}}}$.

Therefore, the probability $\mathbf{e}_1 \in \mathrm{Col}(\mathbf{M})$ is:

$$\begin{aligned} \Pr(\mathbf{e}_1 \in \mathrm{Col}(\mathbf{M})) &= \sum_{k=0}^{N_{u+}} \zeta_k^{r-1,N_{u+}}(1 - \frac{q^k}{q^{N_{u+}}}) \\ &= 1 - \sum_{k=0}^{N_{u+}} \zeta_k^{r-1,N_{u+}} q^{k-N_{u+}}. \end{aligned}$$

Observe that

$$\begin{aligned}\Pr(\mathbf{e}_1 \in \mathrm{Col}(\mathbf{M})) &= \sum_{k=0}^{N_{u+}-1} \zeta_k^{r-1,N_{u+}}(1-\frac{q^k}{q^{N_{u+}}}) \\ &\leq \sum_{k=0}^{N_{u+}-1} \zeta_k^{r-1,N_{u+}} \\ &= 1-\zeta_{N_{u+}}^{r-1,N_{u+}} \\ &= 1-\sum_{i=0}^{N_{u+}-1}(1-q^{-r+1+i}).\end{aligned}$$

Since $\sum_{i=0}^{N_{u+}-1}(1-q^{-r+1+i}) \to 1$ as $q \to \infty$, $\Pr(\mathbf{e}_1 \in \mathrm{Col}(\mathbf{M})) \to 0$, as $q \to \infty$. ■