Rateless Resilient Network Coding Against Byzantine Adversaries

Wentao Huang, Tracey Ho, Hongyi Yao
California Institute of Technology, USA

Sidharth Jaggi
Chinese University of Hong Kong, Hong Kong

Abstract—This paper studies rateless network error correction codes for reliable multicast in the presence of adversarial errors. We present rateless coding schemes for two adversarial models, where the source sends more redundancy over time, until decoding succeeds. The first model assumes there is a secret channel between the source and the destination that the adversaries cannot overhear. The rate of the channel is negligible compared to the main network. In the second model the source and destination share random secrets independent of the input information. The amount of secret information required is negligible compared to the amount of information sent. Both schemes are capacity optimal, distributed, polynomial-time and end-to-end in that other than the source and destination nodes, other intermediate nodes carry out classical random linear network coding.

I. Introduction

Comparing with routing, network coding is more vulnerable to attack by malicious adversaries that inject corrupted packets, since corrupted packets are mixed with other packets in the network. The use of coding to correct such errors information theoretically is studied by [1], [2], [3], [4], [5]. Most existing schemes assume a given min cut (capacity) of the network and maximum number of adversarial errors for the purposes of code design and encoding. However such an assumption may be overly restrictive in many practical settings.

This paper proposes rateless network error correction codes that do not require a priori estimates of the network capacity and number of errors. The source transmits redundancy incrementally until decoding succeeds. The supply of encoded packets is potentially limitless and the number of encoded packets actually transmitted is determined by the number of errors that occur. A number of related works e.g. [6], [7], [8] propose cryptographic schemes that can be used to detect and remove errors in rateless network codes, while [9] proposes a rateless network error correction scheme that requires cryptographic means of verifying successful decoding. In contrast, our work presents the first completely information-theoretic rateless network error correction codes.

We design two algorithms targeting different network models. In the first model, also studied in [4], there is a secret channel between the source and the destination that is hidden from the adversary (who is omniscient except for the secret), and the rate of the channel is negligible compared to the network. In this case over time we incrementally send more linearly dependent redundancy of the source message through the network to combat erasures, and incrementally send more (linearly independent) short hashes of the message on the secret channel to eliminate fake information. The destination amasses both kinds of redundancy until he decodes successfully. The code will adapt to the actual min cut of the network as well as the number of errors.

The second scenario is the random secret model [5], where instead of a secret channel, the source and destination share a "small" fixed random secret that is independent of the input message. The amount of secrets required is again negligible compared to the amount of information sent. Compared to the secret channel model, the challenge is that both linearly dependent and independent redundancy must be sent over the public and unreliable network. Again, we propose codes that will adapt to the network and adversary parameters.

Both schemes are distributed with polynomial-time complexity of design and implementation. They assume no knowledge of the topology and work in both wired and wireless networks. Moreover, implementation involves only slightly modifying the source encoder and destination decoder, while internal nodes use standard random linear network coding.

II. Network Models

A. Adversary Model

The source Alice wishes to communicate reliably with the destination Bob over a general network, where there is a hidden attacker Calvin who wants to disrupt the communication. Calvin is assumed to be able to observe all the transmissions over the network, and know the encoding and decoding schemes at all nodes. Calvin can corrupt transmitted packets or inject erroneous packets. Finally, we assume Calvin to be computationally unbounded. In this paper we discuss two models that limit Calvin's knowledge. For the first model, in addition to the given network, there is a secret channel between Alice and Bob. Information transmitted on this channel cannot be observed or modified by Calvin [4]. However, the rate of the channel is negligible compared to the network. In the second model, we assume the source and destination share a small amount of random secret information that is independent with the input information [5]. Again, the amount of secret information required is negligible compared to the amount of information sent.

B. Network Model

We model the network as a hypergraph where nodes are vertices and hyperedges are directed from the transmitting nodes to the set of the receiving nodes. Let $\mathcal{E}$ be the set of

hyperedges and $\mathcal{T}$ be the set of nodes. Alice and Bob are not assumed to know the capacity of the network as well as the number of errors that the adversary can inject.

Alice encodes her information bits into a batch of b packets by the encoding schemes described in subsequent sections. Each packet contains a sequence of $n + b$ symbols from the finite field $\mathbb{F}_q$. Let matrix $X_0 = \mathbb{F}_q^{b \times (n+b)}$ represent one batch of packets from Alice. We call the communication of one batch of information bits X_0 a session. In the rateless setting, a session may require multiple network transmissions until Bob receives enough redundancy to decode correctly. Assume in general that a session involves N stages, *i.e.*, N uses of the network. During the i-th stage, denote the capacity (min cut from Alice to Bob) of the network as M_i, and the number of errors (min cut from Calvin to Bob) that the adversary injects as z_i. We assume $z_i < M_i$, otherwise the network is completely filled with errors. For any realistic network, M_i is always bounded. For example, let c_i be the number of transmission opportunities at the source during the i-th stage, then $M_i \leq c_i$. For convenience we further assume $c_i \leq \bar{c}$, $\forall i$.

III. Code Construction for Secret Channel Model

A. Encoder

Alice's encoder has a structure similar to [4], but operates in a rateless manner. In each session Alice transmits nb incompressible information symbols from $\mathbb{F}_q$ to Bob. Alice arranges them into a matrix $W \in \mathbb{F}_q^{b \times n}$. Let $X_0 = (W \ I_b)$, where I_b is the identity matrix of dimension b. Alice draws a random matrix $K_1 \in \mathbb{F}_q^{c_1 \times b}$ and encodes $X_1 = K_1 X_0$. X_1 is then sent over a network where intermediate nodes implement random linear coding. In addition, Alice sends a hash of the message through the secret channel. She sets $\alpha_1 = bc_1$, and draws random symbols $r_1, ..., r_{\alpha_1 + 1}$ independently and uniformly from $\mathbb{F}_q$. Note that the $\{r_j\}$ are drawn secretly so that Calvin cannot observe them. Let $D_1 = [d_{kj}] \in \mathbb{F}_q^{(n+b) \times (\alpha_1 + 1)}$, where $d_{kj} = (r_j)^k$, and the hash is computed as $H_1 = X_0 D_1$. Finally Alice sends $r_1, ..., r_{\alpha_1 + 1}$ and H_1 to Bob through the secret channel. The size of the secret is $(\alpha_1 + 1)(b + 1)$, which is asymptotically negligible in n.

Alice keeps sending more redundant information to Bob as follows. For the i-th stage, $i \geq 2$, Alice draws a random matrix $K_i \in \mathbb{F}_q^{c_i \times b}$, encodes $X_i = K_i X_0$, and sends X_i over the network. In addition, Alice again draws $r_1, ..., r_{\alpha_i}$ randomly from $\mathbb{F}_q$ secretly, where $\alpha_i = bc_i$. She then constructs $D_i = [d_{kj}] \in \mathbb{F}_q^{(n+b) \times \alpha_i}$, $d_{kj} = (r_j)^k$, and computes $H_i = X_0 D_i$. Alice eventually sends $r_1, ..., r_{\alpha_i}$ and H_i to Bob through the secret channel. The size of the secret is $\alpha_i(b + 1)$, again asymptotically negligible in n. Note that the secret sent in the first stage is slightly longer in order to guarantee message integrity. Alice repeats this procedure until Bob indicates decoding success. If a success is indicated, Alice ends the current session and moves onto the next session.

B. Decoder

The network performs a classical distributed network code. Specifically, each packet transmitted by an intermediate node

is a random linear combination of its incoming packets. For the i-th stage, we can describe this linear relation as

$$Y_i = [T_i \ Q_i] \begin{bmatrix} X_i \\ Z_i \end{bmatrix},$$

where $Y_i \in \mathbb{F}_q^{M_i \times (n+b)}$ is Bob's received observation, $Z_i \in \mathbb{F}_q^{z_i \times (n+b)}$ represents the errors injected by Calvin, and T_i and Q_i are defined to be the transfer matrix from Alice to Bob and from Calvin to Bob, respectively. By stacking all the batches of observations received by the i-th stage, let

$$Y^{(i)} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_i \end{bmatrix}, \quad Z^{(i)} = \begin{bmatrix} Z_1 \\ \vdots \\ Z_i \end{bmatrix},$$

$$\hat{T}^{(i)} = \begin{bmatrix} T_1 K_1 & Q_1 & 0 & ... & 0 \\ \vdots & \vdots & & & \vdots \\ T_i K_i & 0 & 0 & ... & Q_i \end{bmatrix} = \begin{bmatrix} T^{(i)} \mid Q^{(i)} \end{bmatrix}.$$

and $H^{(i)} = [H_1 ... H_i]$, $D^{(i)} = [D_1 ... D_i]$. Then we have

$$Y^{(i)} = \begin{bmatrix} T^{(i)} & Q^{(i)} \end{bmatrix} \begin{bmatrix} X_0 \\ Z^{(i)} \end{bmatrix}, \tag{1}$$

$$X_0 D^{(i)} = H^{(i)}, \tag{2}$$

where (1) follows from the network transform, and (2) follows from the code construction. Note that only $Y^{(i)}$, $D^{(i)}$ and $H^{(i)}$ are available to Bob, and he needs to recover X_0 from equations (1), (2). To decode, Bob first solve for X^s from

$$X^s Y^{(i)} D^{(i)} = H^{(i)}. \tag{3}$$

If (3) has a unique solution, Bob reconstructs X_0 as following

$$X_0 = X^s Y^{(i)}. \tag{4}$$

Otherwise, as will be shown later, with high probability there is no solution for (3), and Bob waits to receive more redundancy.

C. Performance

In the following we show that the probability of error vanishes as $q \to \infty$. The following Lemma 1 validates that with high probability[1] there exists X^s such that (4) holds.

Lemma 1: If $b + \sum_{j=1}^{i} z_j \leq \sum_{j=1}^{i} M_j$, then $\hat{T}^{(i)}$ has full column rank with high probability.

Proof Sketch: Note $\hat{T}^{(i)} = \begin{bmatrix} T^{(i)} & Q^{(i)} \end{bmatrix}$. $b \leq \sum_{j=1}^{i} M_j$ implies random matrix $T^{(i)}$ has full column rank. So $T^{(i)}$ has full column rank. Without loss of generality we assume $Q^{(i)}$ also has full column rank. Finally by [10], if $b + \sum_{j=1}^{i} z_j \leq \sum_{j=1}^{i} M_j$, the probability that the column spans of $T^{(i)}$ and $Q^{(i)}$ intersects except for the zero vector is upper bounded by $i^2 |\mathcal{T}||\mathcal{E}| q^{-1} \to 0$. Refer to [11] for the details of proof. ∎

Lemma 2: For any $X' \neq X_0$, the probability that $X' D^{(i)} = H^{(i)}$ is bounded from above by $((n+b)/q)^{\sum_{k=1}^{i} \alpha_k + 1}$.

Proof: It is equivalent to consider the probability that $(X' - X_0) D^{(i)} = 0$. Since $X' - X_0 \neq 0$, there is at least one

---

[1]Event E happens with high probability (*w.h.p.*) if $\lim_{q \to \infty} \Pr\{E\} = 1$.

row in which X' differs from X_0. Denote this row of $X' - X_0$ as $(x_1, ..., x_{n+b})$, then the j-th entry of the corresponding row of $(X' - X_0)D^{(i)}$ is $F(r_j) = \sum_{k=1}^{n+b} x_k r_j^k$. Because $F(r_j)$ is not the zero polynomial, the probability (over r_j) that $F(r_j) = 0$ is at most $(n+b)/q$. Because $D^{(i)}$ has $\sum_{k=1}^{i} \alpha_k + 1$ columns, and all r_j, $1 \leq j \leq \sum_{k=1}^{i} \alpha_k + 1$, are independently chosen, the probability that the entire row is a zero vector is at most $((n+b)/q)^{\sum_{k=1}^{i} \alpha_k + 1}$. This is an upper bound on the probability that the entire matrix $(X' - X_0)D^{(i)}$ is zero. ■
Using Lemma 2 and taking the union bound over V^s we have:

Lemma 3: The probability that there exists $V^s \neq X^s$ such that $V^s Y^{(i)} \neq X_0$ but $V^s Y^{(i)} D^{(i)} = H^{(i)}$ is upper bounded by $(n+b)^{\sum_{k=1}^{i} \alpha_k + 1}/q \rightarrow 0$.
Theorem 1 is an immediate consequence of Lemma 1 and 3.

Theorem 1: $\forall i$ such that $b + \sum_{j=1}^{i} z_j \leq \sum_{j=1}^{i} M_j$, Bob decodes X_0 correctly with high probability at the i-th stage. Otherwise, Bob waits for more redundancy.

Theorem 1 shows that the code is optimal in that decoding succeeds with high probability whenever the total amount of information received by the sink satisfies the necessary cut set bound, $b + \sum_{j=1}^{i} z_j \leq \sum_{j=1}^{i} M_j$. The computational cost of design, encoding, and decoding is dominated by the cost of the matrix multiplication $Y^{(i)} D^{(i)}$ in (3), which is $O(n(i\bar{c})^3)$. Details about efficient implementation are available in [11].

IV. CODE CONSTRUCTION FOR RANDOM SECRET MODEL

In this section we assume Alice and Bob share a random secrets whose size is asymptotically negligible compared to the amount of information sent. The shared random secret is assumed to be independent with the source message X_0. Comparing to the previous secret channel model, the random secret model is more challenging because the hashes cannot be computed straightforwardly as in (2), and they must be sent through the public and unreliable network.

The vectorization of a matrix is a linear transformation which converts the matrix into a column vector by stacking the columns of the matrix on top of one another. Let column vector $\boldsymbol{w} \in \mathbb{F}_q^{bn}$ be the vectorized W. To generate hashes, *i.e.*, linearly independent redundancy that is transmitted at the k-th stage, we first draw α_k symbols from the random shared secrets as $d_1^{(k)}, d_2^{(k)}, ..., d_{\alpha_k}^{(k)} \in \mathbb{F}_q$, and use them to construct the $\alpha_k \times nb$ parity check matrix $D_k = \left[d_{ij}^{(k)} \right]$, where $d_{ij}^{(k)} = \left(d_i^{(k)} \right)^j$, $1 \leq i \leq \alpha_k$, $1 \leq j \leq nb$. Then we draw another α_k symbols $\boldsymbol{h}_k = (h_1^{(k)}, ..., h_{\alpha_k}^{(k)})^T$ from the random shared secrets and enforce the following parity check relation:

$$[D_k \ I_{\alpha_k}] \begin{bmatrix} \boldsymbol{w} \\ \boldsymbol{l}_k \end{bmatrix} = \boldsymbol{h}_k, \qquad (5)$$

where I_{α_k} is the identity matrix of dimension α_k and $\boldsymbol{l}_k$ is a vector of length α_k that can be solved for uniquely. So we have a rateless parity check scheme based on (5):

$$\begin{bmatrix} D_1 & I_{\alpha_1} & 0 & ... & 0 \\ D_2 & 0 & I_{\alpha_2} & ... & 0 \\ \vdots & & \vdots & & \vdots \\ D_i & 0 & 0 & ... & I_{\alpha_i} \end{bmatrix} \begin{bmatrix} \boldsymbol{w} \\ \boldsymbol{l}_1 \\ \vdots \\ \boldsymbol{l}_i \end{bmatrix} = \begin{bmatrix} \boldsymbol{h}_1 \\ \vdots \\ \boldsymbol{h}_i \end{bmatrix}, \quad (6)$$

i.e., the total number of parity checks $\sum_i \alpha_i$ can grow over time if necessary.

A. Encoder

In order for Bob to decode successfully, both linearly dependent redundancy and linearly independent redundancy are required. Linearly dependent redundancy corresponds to long messages that lie in the row space of X_0, while the linearly independent redundancy are short hashes with size independent of n. Therefore, it is convenient and efficient to encode and send the two kinds of redundancy separately as long packets and short packets, respectively. We define M_i, z_i, c_i, $\bar{c}$ for long packets as described in Section II. For short packets, denote $\bar{M}_i$, $\bar{c}_i$ and $\bar{z}_i$ as the min cut from Alice to Bob, the number of available transmission opportunities, and the min cut from Calvin to Bob at stage i, respectively. Similarly we assume $\bar{z}_i < \bar{M}_i$, $\forall i$.

The source message is arranged as a $b \times n$ matrix W. Then we let $X_0 = (W \ I_b)$. At the i-th stage, Alice draws a random matrix $K_i \in \mathbb{F}_q^{c_i \times b}$, and encodes the long packets $X_i = K_i X_0$.

To generate the linearly independent redundancy, Alice may choose any σ such that $\sigma \leq \bar{M}_i - \bar{z}_i$, $\forall i$ (e.g., $\sigma = 1$ is a safe choice) and m such that $\sigma m \geq 2b\bar{c} + 2\sigma\bar{c} + 1$. At stage i Alice sets $\alpha_i = i\sigma m$, solves for $\boldsymbol{l}_i$ according to (5), and arranges the column vector into a $\sigma \times im$ matrix $\mathcal{L}_i$. Let $L_j = (\mathcal{L}_j \ \boldsymbol{0}_D \ \boldsymbol{0}_j \ I_\sigma)$, $1 \leq j \leq i$, where $\boldsymbol{0}_D$ is a zero matrix of size $\sigma \times (i-j)m$, and $\boldsymbol{0}_j$ is the zero matrix of size $\sigma \times (j-1)\sigma$. $\boldsymbol{0}_D$ is dummy and is used to align $\mathcal{L}$, and $\boldsymbol{0}_j$ is used to align the identity matrix. Alice then draws a uniform random matrix G_i of size $\bar{c}_i \times i\sigma$ and encodes the short packets as

$$A_i = G_i \begin{bmatrix} L_1 & 0 & ... & 0 \\ \hline L_2 & & ... & 0 \\ & & ... & \\ \hline & & L_i & \end{bmatrix} = G_i L^{(i)}.$$

Note that the size of the secret, $i(i+1)\sigma m/2$, is asymptotically negligible in n. Finally, at the i-th stage Alice sends X_i as long packets and A_i as short packets. Alice repeats this procedure until Bob decodes successfully.

B. Decoder

At stage i Bob receives long and short packets Y_i and J_i:

$$Y_i = T_i X_i + Q_i Z_i, \qquad (7)$$
$$J_i = \bar{T}_i A_i + \bar{Q}_i E_i, \qquad (8)$$

where $T_i \in \mathbb{F}_q^{M_i \times c_i}$, $\bar{T}_i \in \mathbb{F}_q^{\bar{M}_i \times \bar{c}_i}$ are the transfer matrices between Alice and Bob, $Q_i \in \mathbb{F}_q^{M_i \times z_i}$, $\bar{Q}_i \in \mathbb{F}_q^{\bar{M}_i \times \bar{z}_i}$ are the transfer matrices between Calvin and Bob, and $Z_i \in \mathbb{F}_q^{z_i \times (n+b)}$, $E_i \in \mathbb{F}_q^{\bar{z}_i \times i(m+\sigma)}$ are the errors injected to long packets and short packets, respectively. Bob then stacks the long and short packets that he has received so far to get

$$Y^{(i)} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_i \end{bmatrix}, \quad J^{(i)} = \begin{bmatrix} J_1 & 0 & ... & 0 \\ \hline J_2 & & ... & 0 \\ & & ... & \\ \hline & & J_i & \end{bmatrix}.$$

Bob evaluates the rank of $Y^{(i)}$, and waits for more packets until $r_i = \text{Rank}(Y^{(i)}) \geq b$. Without loss of generality we assume the rows of $Y^{(i)}$ are linearly independent. Otherwise, Bob selects r_i linearly independent rows from $Y^{(i)}$ and proceeds similarly. He then picks a basis for the column space of $Y^{(i)}$. As will be shown later, the last b columns of $Y^{(i)}$ are linearly independent $w.h.p.$, so they are chosen, and denoted by an $r_i \times b$ matrix $\hat{T}^{(i)}$. Without loss of generality (by permuting the columns if necessary) we assume that the remaining $r_i - b$ linearly independent columns correspond to the first $r_i - b$ columns of $Y^{(i)}$, denoted by an $r_i \times (r_i - b)$ matrix $T''^{(i)}$. We expand $Y^{(i)}$ with respect to this basis as

$$Y^{(i)} = [T''^{(i)}\ \hat{T}^{(i)}] \begin{bmatrix} I_{r_i-b} & F^Z & 0 \\ 0 & F^X & I_b \end{bmatrix}, \qquad (9)$$

where F^Z and F^X are matrices of coefficients.

Bob deals with $J^{(i)}$ in a similar way. Let $\bar{r}_i$ be the rank of $J^{(i)}$, $\hat{\bar{T}}^{(i)} \in \mathbb{F}_q^{\bar{r}_i \times i\sigma}$ be the last $i\sigma$ columns of $J^{(i)}$, and $\bar{T}''^{(i)} \in \mathbb{F}_q^{\bar{r}_i \times (\bar{r}_i - i\sigma)}$ be the first $\bar{r}_i - i\sigma$ columns of $J^{(i)}$. Then $w.h.p.$ $[\bar{T}''^{(i)}\ \hat{\bar{T}}^{(i)}]$ comprises a basis for the column space of $J^{(i)}$, and we can write

$$J^{(i)} = [\bar{T}''^{(i)}\ \hat{\bar{T}}^{(i)}] \begin{bmatrix} I_{\bar{r}_i-i\sigma} & F^E & 0 \\ 0 & F^A & I_{i\sigma} \end{bmatrix}. \qquad (10)$$

Equations (9) and (10) characterize the effect of the network transform. To take into account the built-in redundancy of the message, $\forall i$, Bob splits X_0 and $L^{(i)}$ as:

$$X_0 = [X_a^{(i)}\ X_b^{(i)}\ X_c^{(i)}], \qquad (11)$$

$$L^{(i)} = [L_a^{(i)}\ L_b^{(i)}\ L_c^{(i)}], \qquad (12)$$

where $X_a^{(i)}$ are the first $r_i - b$ columns of X_0, $X_c^{(i)}$ are the last b columns of X_0, and $X_b^{(i)}$ are the remaining columns in the middle; $L_a^{(i)}$ are the first $\bar{r}_i - i\sigma$ columns of $L^{(i)}$, $L_c^{(i)}$ are the last $i\sigma$ columns of $L^{(i)}$, and $L_b^{(i)}$ are the remaining columns in the middle. Let $\boldsymbol{x}_a^{(i)}$, $\boldsymbol{x}_b^{(i)}$ and $\boldsymbol{x}_c^{(i)}$ be the vectorized versions of $X_a^{(i)}$, $X_b^{(i)}$ and $X_c^{(i)}$. Let $\boldsymbol{l}_a^{(i)}$, $\boldsymbol{l}_b^{(i)}$ and $\boldsymbol{l}_c^{(i)}$ be the vectorized versions of $L_a^{(i)}$, $L_b^{(i)}$ and $L_c^{(i)}$ omitting the dummy $\boldsymbol{0}_D$. By construction it follows that,

$$\begin{bmatrix} \boldsymbol{x}_a^{(i)} \\ \boldsymbol{x}_b^{(i)} \\ \hline \boldsymbol{l}_a^{(i)} \\ \boldsymbol{l}_b^{(i)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{w} \\ \hline \boldsymbol{l}_1 \\ \vdots \\ \boldsymbol{l}_i \end{bmatrix}. \qquad (13)$$

Then Bob constructs two matrices B_{top} and B_{mid} as defined in (14) and (15), respectively. Here $f_{i,j}^Z$ and $f_{i,j}^E$ are the $(i,j)^{th}$ entries of matrix F^Z and F^E, and $\beta = n + b - r_i$, $\gamma = i(m+\sigma) - \bar{r}_i$. Let the j-th column of B_{mid} corresponds to the j-th entry of the vectorized $L^{(i)}$. Bob deletes from B_{mid} all columns corresponding to dummy zero paddings in $L^{(i)}$, and obtains a submatrix B'_{mid}.

$$B_{top} = \left[\begin{array}{ccc|ccc} -f_{1,1}^Z \hat{T}^{(i)} & \cdots & -f_{r_i-b,1}^Z \hat{T}^{(i)} & \hat{T}^{(i)} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -f_{1,\beta}^Z \hat{T}^{(i)} & \cdots & -f_{r_i-b,\beta}^Z \hat{T}^{(i)} & 0 & \cdots & \hat{T}^{(i)} \end{array} \right] \qquad (14)$$

$$B_{mid} = \left[\begin{array}{ccc|ccc} -f_{1,1}^E \hat{\bar{T}}^{(i)} & \cdots & -f_{\bar{r}_i-i\sigma,1}^E \hat{\bar{T}}^{(i)} & \hat{\bar{T}}^{(i)} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -f_{1,\gamma}^E \hat{\bar{T}}^{(i)} & \cdots & -f_{\bar{r}_i-i\sigma,\gamma}^E \hat{\bar{T}}^{(i)} & 0 & \cdots & \hat{\bar{T}}^{(i)} \end{array} \right] \qquad (15)$$

Finally, Bob let

$$B_{bot} = \begin{bmatrix} D_1 & I_{\alpha_1} & \cdots & 0 \\ \vdots & & \vdots & \\ D_i & 0 & \cdots & I_{\alpha_i} \end{bmatrix},$$

If Bob permutes the columns of $Y^{(i)}$ and $J^{(i)}$ when constructing $T''^{(i)}$ and $\bar{T}''^{(i)}$, then he needs to permute the columns of B_{bot} accordingly. Then he tries to solve the equations:

$$B \begin{bmatrix} \boldsymbol{x}_a^{(i)} \\ \boldsymbol{x}_b^{(i)} \\ \boldsymbol{l}_a^{(i)} \\ \boldsymbol{l}_b^{(i)} \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{T}}^{(i)} \boldsymbol{f}^X \\ \hat{\bar{\boldsymbol{T}}}^{(i)} \boldsymbol{f}^A \\ \boldsymbol{h}_1 \\ \vdots \\ \boldsymbol{h}_i \end{bmatrix}, \qquad (16)$$

where $\boldsymbol{f}^X$, $\boldsymbol{f}^A$ are the vectorized versions of F^X, F^A, respectively, $\hat{\boldsymbol{T}}^{(i)} = \text{diag}[\hat{T}^{(i)}, ..., \hat{T}^{(i)}]$, $\hat{\bar{\boldsymbol{T}}}^{(i)} = \text{diag}[\hat{\bar{T}}^{(i)}, ..., \hat{\bar{T}}^{(i)}]$, and the matrix B is defined as:

$$B = \left[\begin{array}{c|c} B_{top} & 0 \\ 0 & B'_{mid} \\ \hline \multicolumn{2}{c}{B_{bot}} \end{array} \right].$$

Bob tries to solve (16) and if there is a unique solution, Bob has decoded successfully with high probability. Otherwise, with high probability there is no solution and Bob waits to receive more redundancy.

C. Performance

Again we will show the probability of error is vanishing. The following Lemmas 4 and 5 establish (10) and (9). The idea of their proofs is similar to Lemma 1.

Lemma 4: $\hat{\bar{T}}^{(i)}$ has full column rank with high probability.

Lemma 5: If $\sum_{j=1}^i M_j - \sum_{j=1}^i z_j \geq b$, then $\hat{T}^{(i)}$ has full column rank with high probability.

The following Lemmas 6 and 7 can be proved by standard matrix operations and by invoking Lemmas 4 and 5. We defer detail proofs to [11] due to space limit.

Lemma 6: With high probability (8) and (10) are equivalent to the following equation:

$$\hat{\bar{T}}^{(i)} L_b^{(i)} = \hat{\bar{T}}^{(i)} (F^A + L_a^{(i)} F^E). \qquad (17)$$

Lemma 7: If $\sum_{j=1}^i M_j - \sum_{j=1}^i z_j \geq b$, then with high probability (7) and (9) are equivalent to

$$\hat{T}^{(i)} X_b^{(i)} = \hat{T}^{(i)} (F^X + X_a^{(i)} F^Z). \qquad (18)$$

Equations (6), (18), and (17) together imply:

Corollary 1: If $\sum_{j=1}^i M_j - \sum_{j=1}^i z_j \geq b$, then the matrix equation (16) holds with high probability.

Finally we need to prove that (16) has a unique solution.

Lemma 8: If $\sigma m \geq 2b\bar{c} + 2\sigma\bar{c} + 1$, then with high probability there does not exist $X' \neq X_0$ such that X' satisfies (16).

Proof: Suppose $X' \neq X_0$, and let $\boldsymbol{x}'_a$, $\boldsymbol{x}'_b$ be the corresponding vectorized components as in (11). We study the probability that there exist $\boldsymbol{x}'_a$, $\boldsymbol{x}'_b$, $\boldsymbol{l}'_a$ and $\boldsymbol{l}'_b$ that satisfy (16). Consider the top $\beta r_i + \gamma \bar{r}_i$ rows in B corresponding to the blocks of B_{top} and B'_{mid}

$$\begin{bmatrix} B_{top} & \mathbf{0} \\ \mathbf{0} & B'_{mid} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}'^{(i)}_a \\ \boldsymbol{x}'^{(i)}_b \\ \boldsymbol{l}'^{(i)}_a \\ \boldsymbol{l}'^{(i)}_b \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{T}}^{(i)} \boldsymbol{f}^X \\ \hat{\boldsymbol{T}}^{(i)} \boldsymbol{f}^A \end{bmatrix}, \quad (19)$$

They are equivalent to

$$X'^{(i)}_b = F^X + X'^{(i)}_a F^Z \quad (20)$$
$$L'^{(i)}_b = F^A + L'^{(i)}_a F^E \quad (21)$$

Therefore given arbitrary values of $\boldsymbol{x}'^{(i)}_a$ and $\boldsymbol{l}'^{(i)}_a$, there are unique corresponding values of $\boldsymbol{x}'^{(i)}_b$ and $\boldsymbol{l}'^{(i)}_b$ that satisfy (19).

Now given any $\boldsymbol{x}'^{(i)}_a$ and $\boldsymbol{l}'^{(i)}_a$ (and the corresponding $\boldsymbol{x}'^{(i)}_b$ and $\boldsymbol{l}'^{(i)}_b$) such that (19) holds, we consider the probability that the bottom $\sum_{k=1}^{i} \alpha_k = (i^2 + i)\sigma m/2$ rows in (16) also hold:

$$B_{bot} \begin{bmatrix} \boldsymbol{x}'^{(i)}_a \\ \boldsymbol{x}'^{(i)}_b \\ \boldsymbol{l}'^{(i)}_a \\ \boldsymbol{l}'^{(i)}_b \end{bmatrix} = \begin{bmatrix} \boldsymbol{h_1} \\ \vdots \\ \boldsymbol{h_i} \end{bmatrix} \quad (22)$$

This is equivalent to:

$$B_{bot} \begin{bmatrix} \boldsymbol{x}^{(i)}_a - \boldsymbol{x}'^{(i)}_a \\ \boldsymbol{x}^{(i)}_b - \boldsymbol{x}'^{(i)}_b \\ \boldsymbol{l}^{(i)}_a - \boldsymbol{l}'^{(i)}_a \\ \boldsymbol{l}^{(i)}_b - \boldsymbol{l}'^{(i)}_b \end{bmatrix} = \mathbf{0}, \quad (23)$$

Because $X' \neq X_0$, so $\boldsymbol{x}^{(i)}_a - \boldsymbol{x}'^{(i)}_a$ and $\boldsymbol{x}^{(i)}_b - \boldsymbol{x}'^{(i)}_b$ cannot both be the zero vector. Denote

$$\boldsymbol{x}^{(i)}_a - \boldsymbol{x}'^{(i)}_a = (x^{(i)}_{a,1}, ..., x^{(i)}_{a,\theta_a})^T$$
$$\boldsymbol{x}^{(i)}_b - \boldsymbol{x}'^{(i)}_b = (x^{(i)}_{b,1}, ..., x^{(i)}_{b,\theta_b})^T$$
$$\begin{bmatrix} \boldsymbol{l}^{(i)}_a - \boldsymbol{l}'^{(i)}_a \\ \boldsymbol{l}^{(i)}_b - \boldsymbol{l}'^{(i)}_b \end{bmatrix} = (l^{(i)}_1, ..., l^{(i)}_{\theta_l})^T$$

where $\theta_a = b(r_i - b)$, $\theta_b = \beta b$ and $\theta_l = (i^2 + i)\sigma m/2$. Denote the (u, v) entry of B_{bot} as $s_{u,v}$, then the j-th row of (23) is

$$\sum_{k=1}^{b(r_i-b)} x^{(i)}_{a,k} s_{j,k} + \sum_{k=1}^{\beta b} x^{(i)}_{b,k} s_{j,k+b(r_i-b)}$$
$$+ \sum_{k=1}^{(i^2+i)\sigma m/2} l^{(i)}_k s_{j,k+nb} = 0 \quad (24)$$

Let s_j be the $(j, 1)$ entry of B_{bot} before column permutation, then $s_{j,k} = s_j^{\pi(k)}$, $1 \leq k \leq nb$, where π is a permutation of $\{1, ..., nb\}$. So (24) is a non-zero polynomial of order at most $b(r_i - b) + \beta b = nb$ in variable s_j (the $\{s_{j,k+nb}\}$ are constants 0 or 1 by construction and are independent with respect to

s_j). By the fundamental theorem of algebra the polynomial has at most nb roots. The probability that s_j is chosen as one of the roots is at most nb/q, and this is the upper bound of the probability that row j holds in (23). Because $\{s_j\}$ are chosen independently, (23) holds with probability no larger than $(nb/q)^{(i^2+i)\sigma m/2}$.

Finally, there are at most $q^{b(r_i-b)}$ different $\boldsymbol{x}^{(i)}_a$ and at most $q^{i\sigma(\bar{r}_i-i\sigma)}$ different $\boldsymbol{l}^{(i)}_a$. By (7), $r_i - b \leq i\bar{c}$, and by (8), $\bar{r}_i - i\sigma \leq i\bar{c}$. By the union bound, the probability that there exists $X'_0 \neq X_0$ such that $\boldsymbol{x}'_a$, $\boldsymbol{x}'_b$, $\boldsymbol{l}'_a$ and $\boldsymbol{l}'_b$ satisfy (16) is at most

$$\left(\frac{nb}{q}\right)^{\frac{(i^2+i)\sigma m}{2}} q^{ib\bar{c}+i^2\sigma\bar{c}} \leq \frac{(nb)^{i^2\sigma m}}{q^{i^2}} \to 0$$

$\blacksquare$

We are ready to present the final conclusion.

Theorem 2: $\forall i$ such that $b + \sum_{j=1}^{i} z_j \leq \sum_{j=1}^{i} M_j$, with the proposed coding scheme, Bob is able to decode X_0 correctly with high probability at the i-th stage. Otherwise, Bob waits for more redundancy instead of decoding erroneous packets.

Proof: By Corollary 1, X_0 can be solved from (16) if $b + \sum_{j=1}^{i} z_j \leq \sum_{j=1}^{i} M_j$. By Lemma 8, if a solution exists, it is correct and unique. Otherwise, there is no solution to (16) and by the algorithm Bob waits for more redundancy. $\blacksquare$

Theorem 2 shows that our code is optimal in that sense that decoding succeeds with high probability whenever the total amount of information received by the sink satisfies the cut set bound with respect to the amount of message and error information. The computational cost of design, encoding, and decoding is dominated by the cost of solving (16), which equals $O((ni\bar{c})^3)$.

REFERENCES

[1] R. W. Yeung and N. Cai, "Network error correction, part i: Basic concepts and upper bounds," *Communications in Information and Systems*, vol. 6, no. 1, pp. 19 –36, 2006.
[2] N. Cai and R. W. Yeung, "Network error correction, part ii: Lower bounds," *Communications in Information and Systems*, vol. 6, no. 1, pp. 37 – 54, 2006.
[3] R. Koetter and F. R. Kschischang, "Coding for errors and erasures in random network coding," *IEEE Trans. Info. Theory*, August 2008.
[4] S. Jaggi, M. Langberg, S. Katti, T. Ho, D. Katabi, M. Medard, and M. Effros, "Resilient network coding in the presence of byzantine adversaries," *IEEE Trans. Info. Theory*, 2008.
[5] L. Nutman and M. Langberg, "Adversarial models and resilient schemes for network coding," in *IEEE ISIT*, 2008, pp. 171–175.
[6] M. N. Krohn, M. J. Freedman, and D. Mazieres, "On-the-fly verification of rateless erasure codes for efficient content distribution," in *IEEE Symp. Security and Privacy*, 2004, pp. 226–240.
[7] C. Gkantsidis and P. Rodriguez Rodriguez, "Cooperative security for network coding file distribution," in *IEEE INFOCOM*, 2006, pp. 1–13.
[8] D. Charles, K. Jain, and K. Lauter, "Signatures for network coding," in *Proc. 40th Annual Conf. Info. Sciences and Systems*, 2006, pp. 857–863.
[9] S. Vyetrenko, A. Khosla, and T. Ho, "On combining information-theoretic and cryptographic approaches to network coding security against the pollution attack," in *Asilomar Conf.*, 2009, pp. 788–792.
[10] T. Ho, M. Mdard, J. Shi, M. Effros, and D. Karger, "On randomized network coding," in *in Proc. Allerton Conf.*, Monticello, IL, 2003.
[11] W. Huang, T. Ho, H. Yao, and S. Jaggi, "Rateless resilient network coding against byzantine adversaries," *Arxiv:1301.2860*, Jan 2013.