

Network error correction with unequal link capacities

Sukwon Kim, Tracey Ho, Michelle Effros, and Salman Avestimehr

Abstract—We study network error correction with unequal link capacities. Previous results on network error correction assume unit link capacities. We consider network error correction codes that can correct arbitrary errors occurring on up to z links. We find the capacity of a network consisting of parallel links, and a generalized Singleton outer bound for any arbitrary network. We show by example that linear coding is insufficient for achieving capacity in general. In our example, the capacity is 50% greater than the linear coding capacity and we achieve using a nonlinear error detection strategy. We also present a method for finding an upper bound on the linear coding capacity for arbitrary network. We show that even for a single source and single sink network, it may be necessary for intermediate nodes to do coding, nonlinear error detection or error correction. This is unlike the equal link capacity case, where coding only at the source and forwarding at intermediate nodes suffices for a single source and sink network. We conjecture that the generalized Singleton outer bound is not achievable in general.

I. INTRODUCTION

Recently, Yeung and Cai introduced the network error-correction problem [1], [2]. They generalized the Hamming bound, the Singleton bound, and the Gilbert-Varshamov bound from classical error correction coding to network coding. In [3], Zhang introduced the concept of minimum rank for linear network codes, which plays a similar role in classical error-correcting codes. In [4], [5], the authors presented a refined version of the Singleton bound for network error correction.

In all previous works on network error correction, the authors assume unit link capacity. A t -error-correcting code is defined as follows: if the total number of links in the network that may be corrupted by errors is at most t , then the source message can be recovered by all sink nodes. In this scenario, the tightness of the Singleton bound is proved, and linear network error-correcting codes are optimal in the sense of t -error-correction [2, Theorem 4]. The capacity is $C - 2t$, where C is the min-cut of the network.

In the error-free case, any link l with capacity r can be represented by r edges of capacity one without loss of generality. However, in the case with errors, there is a loss of generality in assuming that errors occur independently on the unit capacity edges.

Department of Electrical Engineering, California Institute of Technology, Pasadena, CA, 91125, USA, {sukwon, tho, effros, avestimeh}@caltech.edu.

This work is partly supported under a subcontract #069153 issued by BAE Systems National Security Solutions, Inc. and supported by the Defense Advanced Research Projects Agency (DARPA) and the Space and Naval Warfare System Center (SPAWARSYSCEN), San Diego under Contract No. N66001-08-C-2013.

In this paper, we consider network error correction with unequal link capacities. For any link l in the network with capacity r , r symbols can be transmitted on l . If an adversary controls this link, some or all of the symbols transmitted across the link may be corrupted. We therefore define a z -error link-correcting code to be a code that can recover the source message at all sink nodes if there are at most z adversarial links in the network. In Section III, we study a two-node network with parallel channels of varying capacities. When this network consists of K links of arbitrary capacity and there are at most z adversarial links, we show that the capacity of the network is the sum of the $(K - 2z)$ smallest link capacities.

In Section IV, we extend the parallel channel result to obtain a generalized Singleton outer bound that applies to all network scenarios. In this case, we prove the insufficiency of linear network codes to achieve the capacity in general. The proof uses an example for which we can show that the capacity is 50% greater than the linear coding capacity and we achieve using nonlinear error detection strategy. We generalize our proof and present a method for finding an upper bound on the linear coding capacity for arbitrary network. In Section V, we show that even for a single source and single sink network, it may be necessary for intermediate nodes to do coding, nonlinear error detection or error correction. This is unlike the equal link capacity case, where coding only at the source and forwarding at intermediate nodes suffices for a single source and sink network. We conjecture that the generalized Singleton outer bound is not achievable in general. We provide an example with some intuition. Section VI concludes this paper and proposes future works.

II. PRELIMINARIES

We consider an acyclic communication network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Source node $s \in \mathcal{V}$ transmits information to the sink nodes $u \in \mathcal{U}$. We use $r(a, b)$ to denote the capacity of edge $(a, b) \in \mathcal{E}$. We assume that the code alphabet \mathcal{X} is equal to $GF(q)$ for some prime power q . We regard an error vector in each link $l \in \mathcal{E}$ as set of $r(l)$ symbols in code alphabet \mathcal{X} , with the output y_l of link l equals the modulo q sum of the input x_l to link l and the error e_l applied to link l . We say that τ error links occur in the network if $e_l \neq 0$ on τ links.

Definition 1: A network code is z -error link-correcting if it can correct any τ adversarial links for $\tau \leq z$. That is, if the total number of adversarial links in the network is at most z , then the source message can be recovered by all the sink nodes $u \in \mathcal{U}$.

As in [1], [2], we consider any linear coded multi-cast (LCM) V that assigns a linear subspace $L_v(a)$ to each node $a \in V$ and a set of $r(l)$ column vectors $\{v^\tau(l)_1, v^\tau(l)_2, \dots, v^\tau(l)_{r(l)}\}$ to each link $l \in \mathcal{E}$ in the network. Denote by G_a the matrix whose columns are the vectors assigned to the input links of node a . For any LCM V , there exists a set of $r(l)$ column vectors $\{c^\tau(l)_1, c^\tau(l)_2, \dots, c^\tau(l)_{r(l)}\}$ such that $v^\tau(l)_i = G_a c^\tau(l)_i$. Then we can define a linear network code ϕ based on any LCM V as in [1]. Let

$$\tilde{\phi}_l(w) = \{\langle w, v^\tau(l)_i \rangle : 1 \leq i \leq r(l)\}$$

denote the error-free output of link l when the network input is w . We again use vector e_l to denote the errors on link l and $e = (e_l : l \in \mathcal{E})$ to denote the entire network error. If an error vector e occurs, its components are added to the link inputs according to the coding order. Then the output of a link l is a function of both the network input w and the error vector e . We denote that output by $\psi_l(w, e)$. With this notation, a sink node u cannot distinguish between the case where w is the network input and error e occurs and the case where w' is the network input and error e' occurs if and only if

$$(\psi_l(w, e) : l \in \Gamma_+(u)) = (\psi_l(w', e') : l \in \Gamma_+(u)), \quad (1)$$

where $\Gamma_+(a) = \{(c, a) : (c, a) \in \mathcal{E}\}$ and $\Gamma_-(a) = \{(a, b) : (a, b) \in \mathcal{E}\}$ denote the sets of incoming and outgoing edges of node a , respectively. Let $N(e) = |\{l \in \mathcal{E} : e_l \neq 0\}|$ denote the number of links in which an error occurs. We say that any pair of input vectors w and w' are z links separable at sink node u if (1) does not hold for any pair of error vectors e and e' such that $N(e) \leq z$ and $N(e') \leq z$. Lemma 1 in [2] for the network with unit link capacity can be directly extended to the following lemma for the network with arbitrary link capacity.

Lemma 1: For all $l \in \mathcal{E}$, all network inputs w and w' , error vectors e and e' , and $\mu \in GF(q)$,

$$\psi_l(w + w', e + e') = \psi_l(w, e) + \psi_l(w', e')$$

and

$$\psi_l(\mu w) = \mu \psi_l(w).$$

From Lemma 1,

$$\psi_l(w, e) = \psi_l(w, 0) + \psi_l(0, e) = \tilde{\phi}_l(w) + \theta_l(e).$$

In other words, $\psi_l(w, e)$ can be written as the sum of a linear function of w and a linear function of e .

Let (A, B) be a partition of \mathcal{V} , and define the cut for the partition (A, B) by

$$\text{cut}(A, B) = \{(a, b) \in \mathcal{E} : a \in A, b \in B\}.$$

The quantity $m(A) = \sum_{(a,b) \in \text{cut}(A,B)} r(a,b)$ is called the volume of $\text{cut}(A, B)$. $\text{cut}(A, B)$ is called a cut between two nodes a and b if $a \in A$ and $b \in B$. Let $CS(a, b)$ denote the set of cuts between a and b and let $c(a, b)$ denote the minimum volume of a cut between a and b .

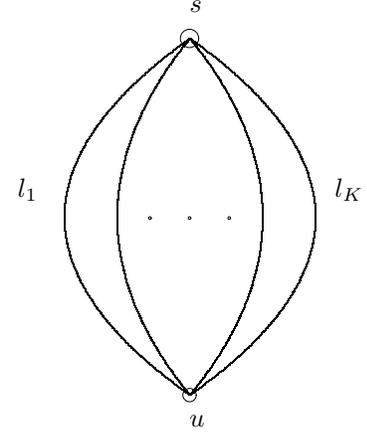


Fig. 1. Two-node network G with K parallel links : there are k_i links of capacity c_i connecting s and u ($c_1 < c_2 < \dots < c_m$).

III. TWO-NODE NETWORK WITH PARALLEL CHANNELS

In this section, we consider a two-node network G with K parallel links between the single source s and sink u , as shown in Fig. 1. There are k_i links of capacity c_i connecting s and u ($c_1 < c_2 < \dots < c_m$). We define the min-cut n as the sum of link capacities, i.e., $n = \sum_{i=1}^m c_i k_i$. Under the assumption that there are at most z adversarial links in this network, we derive the capacity of the network.

Theorem 1: The capacity C is equal to the sum of $K - 2z$ smallest link capacities.

Proof: First we show that C is achievable. We decompose G into m subnetworks (G_1, \dots, G_m) where G_i consists of $\sum_{j=i}^m k_j$ links of capacity $c_i - c_{i-1}$, as shown in Fig. 2. ($c_0 = 0$). Each of G_i contains at most z adversarial links. Since G_i consists of equal capacity links, we can apply the achievability result in [2, Theorem 4]. Suppose that

$$k_p + k_{p+1} + \dots + k_m \geq 2z > k_{p+1} + \dots + k_m$$

for some $1 \leq p \leq m$.

Then, for sufficiently large q , the achievable capacity C for this network is

$$\begin{aligned} & \sum_{i=1}^m (c_i - c_{i-1}) \max(0, \sum_{j=i}^m k_j - 2z) \\ &= \sum_{i=1}^p (c_i - c_{i-1}) (\sum_{j=i}^m k_j - 2z) \\ &= \sum_{i=1}^{p-1} c_i k_i + c_p (k_p + \dots + k_m - 2z) \\ &= C. \end{aligned}$$

As shown above, C is achievable.

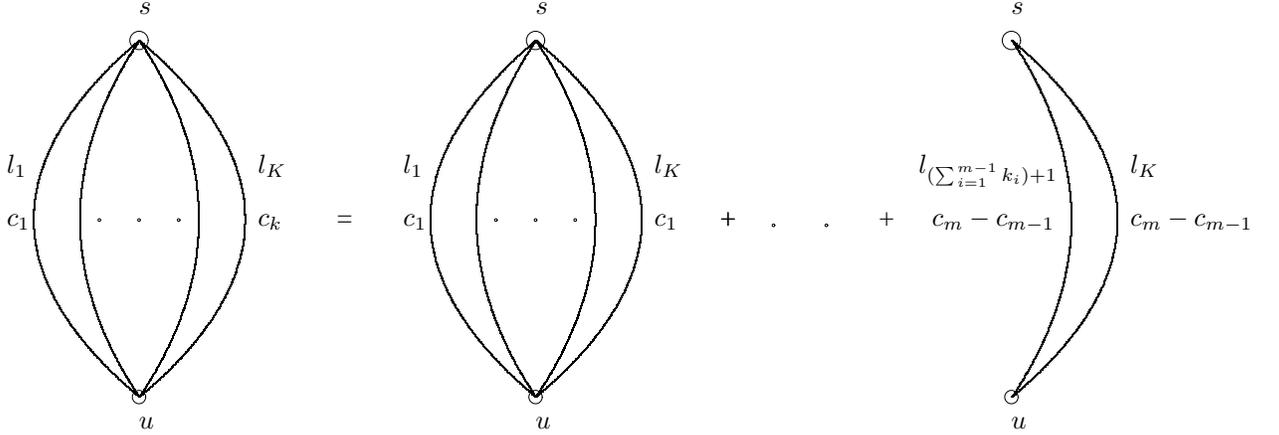


Fig. 2. We decompose G into m subnetworks (G_1, \dots, G_m) where G_i consists of $\sum_{j=i}^m k_j$ links of capacity $c_i - c_{i-1}$.

Now we show the converse. We use \mathcal{X} to denote the source alphabet. If there exists z -error links-correcting code on the two-node network G with source alphabet \mathcal{X} , then

$$\log |\mathcal{X}| \leq C \cdot \log q.$$

We assume that links $\mathcal{E} = \{l_1, \dots, l_K\}$ are indexed in increasing order, i.e., $r(l_1) < \dots < r(l_K)$.

Let $\{\phi_l : l \in \mathcal{E}\}$ be a z -error link-correcting network code transmitting an information from source with alphabet \mathcal{Z} . $\tilde{\phi}_l(x)$ denotes the error-free output of input x on link l . Now we assume that $|\mathcal{X}| > q^C$ and will show that this leads to a contradiction. We use $O(x) = (\tilde{\phi}_{l_1}(x), \dots, \tilde{\phi}_{l_K}(x))$ to denote the error free output of the network with input x . From the definition, C is the sum of the $(K - 2z)$ smallest capacities of links in the network, i.e., $C = \sum_{i=1}^{K-2z} r(l_i)$. From our assumption $|\mathcal{X}| > q^C$, there exist two distinct symbols $x, x' \in X$ such that $\phi_{l_i}(x) = \phi_{l_i}(x') \forall i = 1, \dots, K - 2z$. So we can write

$$O(x) = \{y_1, \dots, y_{K-2z}, u_1, \dots, u_z, w_1, \dots, w_z\},$$

$$O(x') = \{y_1, \dots, y_{K-2z}, u'_1, \dots, u'_z, w'_1, \dots, w'_z\}.$$

We can construct z -error links that changes $O(x)$ to the value $\{y_1, \dots, y_{K-2z}, u'_1, \dots, u'_z, w_1, \dots, w_z\}$ as follows. We apply an error of value $(u'_i - u_i) \bmod q$ on links l_{K-2z+i} for $1 \leq i \leq z$. Since this does not change the output value of other $K - z$ links, we obtain $\{y_1, \dots, y_{K-2z}, u'_1, \dots, u'_z, w_1, \dots, w_z\}$. For the source symbol x' , we can follow a similar procedure to construct z error links that change the value of $O(x')$ to $\{y_1, \dots, y_{K-2z}, u'_1, \dots, u'_z, w_1, \dots, w_z\}$. Thus, sink node u cannot reliably distinguish between the source symbol x and x' , which gives a contradiction. ■

Theorem 1 gives us the capacity of any two-node network with parallel links of varying capacities.

IV. INSUFFICIENCY OF LINEAR NETWORK CODE

Here we first extend the converse part of the two-node network result to obtain a generalized Singleton bound that applies to all network scenarios. We show by example that linear coding is insufficient for achieving capacity in general. In our example, the capacity is 50% greater than the linear coding capacity and we achieve using a nonlinear error detection strategy. We generalize our proof and present a strategy that gives an upper bound on the linear coding capacity for arbitrary network.

A. Generalized Singleton bound

On an acyclic directed graph \mathcal{G} we can define a coding order such that lower-indexed arcs are upstream of higher-indexed arcs. We regard the errors in the network as being injected by a jammer according to the coding order, as in [1].

For a cut $Q \in \text{cut}(s, u)$, let $K(Q)$ denote the number of links in Q .

Lemma 2: Consider any z -error correcting network code $\{\phi_{l_i} : l_i \in E, 1 \leq i \leq K\}$ with source alphabet X . Consider any set S consisting of $2z$ links on a source-sink cut Q such that none of the remaining links on Q are downstream of any link in S . Let M be the total capacity of the remaining links. Then

$$\log |X| \leq M \cdot \log q.$$

Proof: We assume that $|X| > q^M$, and show that this leads to a contradiction.

For brevity, let $Q = \{l_1, \dots, l_{K(Q)}\}$ where $S = \{l_{K(Q)-2z+1}, \dots, l_{K(Q)}\}$ and links in S are in the coding order of the given network code. Since $|X| > q^M$, from the definition of M , there exist two distinct symbols $x, x' \in X$ such that $\phi'_{l_i}(x) = \phi'_{l_i}(x') \forall i = 1, \dots, K(Q) - 2z$. So we can

write

$$O(x) = \{y_1, \dots, y_{K(Q)-2z}, u_1, \dots, u_z, w_1, \dots, w_z\},$$

$$O(x') = \{y_1, \dots, y_{K(Q)-2z}, u'_1, \dots, u'_z, w'_1, \dots, w'_z\}.$$

We will show that it is possible for the jammer to produce exactly the same outputs at all the channels on Q when errors are occurred at most z links on Q .

Assume the input of network is x . The jammer will inject errors on z links $l_{K(Q)-2z+1}, \dots, l_{K(Q)-z}$ in this order as follows. First the jammer applies an error on link $l_{K(Q)-2z+1}$ to change the output from u_1 to u'_1 . Then the output of links $(l_{K(Q)-2z+2}, \dots, l_{K(Q)})$ may be affected, but not the outputs of links $(l_1, \dots, l_{K(Q)-2z})$. Let $u'_i(j)$ and $w'_i(j)$ denote the outputs of links $l_{K(Q)-2z+i}$ and $l_{K(Q)-z+i}$, respectively after the jammer has injected errors on link $l_{K(Q)-2z+j}$, where $j = 1, 2, \dots, t$ with $u'_1(1) = u'_1$. Then the jammer injects errors on link $l_{K(Q)-2z+2}$ to change its output from u'_1 to u_2 . This process continues until the jammer finishes injecting errors on z links $l_{K(Q)-2z+1}, \dots, l_{K(Q)-z}$ and the output of this channel changes from $O(x)$ to $\{y_1, \dots, y_{K(Q)-2z}, u'_1, \dots, u'_z, w'_1(t), \dots, w'_z(t)\}$. Now suppose the input is x' . We can follow a similar procedure by injecting errors on z links $l_{K(Q)-z+1}, \dots, l_{K(Q)}$. Then the jammer can produce the outputs

$$\{y_1, \dots, y_{K(Q)-2z}, u'_1, \dots, u'_z, w'_1(t), \dots, w'_z(t)\}.$$

Thus, sink node u cannot reliably distinguish between the source symbol x and x' , which gives a contradiction. ■

From above Lemma, we derive the cut-set outer bound as follows. Given a cut Q , we consider all possible set S consisting of $2z$ links on the Q such that none of the remaining links on Q are downstream of any link in S . Then we choose a set S^* among them that has the maximum total link capacities. We define $M(Q)$ to be the sum of the capacities of the links on Q which are not in S^* . This gives the outer bound

$$\log |X| \leq \min_{u \in \mathcal{U}} \min_{Q \in \text{cut}(s,u)} M(Q) \cdot \log q.$$

B. Insufficiency of linear network code

We consider a single source-destination network in Fig. 3. Source s transmits the information to a sink node u . We index the links and assume the capacities of links as shown in Fig. 3. For a single adversarial link, generalized Singleton bound is $\min_{\Omega \in \mathcal{C}S(s,u)} M(\Omega) = 2$.

Theorem 2: Given a network in Fig. 3, for a single adversarial link, rate 2 is asymptotically achievable with nonlinear error detection strategy, whereas scalar linear network code achieves at most $4/3$.

Proof: We first illustrate the nonlinear error detection strategy as follows. Source wants to transmit two packets (X, Y) . We send them in n channel uses, but each packet has only $n - 1$ bits. We use one bit as a signaling bit. We send (X, Y) down all links in the top layer. In the middle layer, we do the following operations:

(1) Send the linear combination of X and Y , $aX + bY$, down link l_4 .

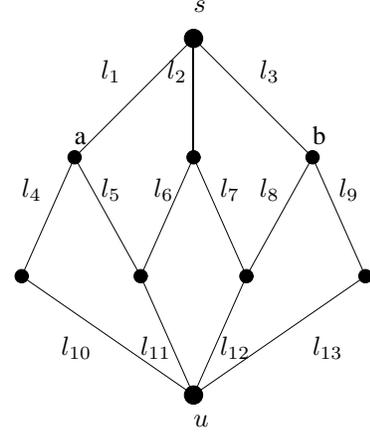


Fig. 3. All links on the top layer have capacity 2. All links on the middle and bottom layer have capacity 1.

- (2) Send X down both links l_5 and l_6 .
- (3) Send Y down both links l_7 and l_8 .
- (4) Send a different linear combination of X and Y , $cX + dY$, down link l_9 .

At the bottom layer, we do the following operations:

- (1) Forward the received packet on link l_{10} .
- (2) Send a 1 followed by X on link l_{11} if the two copies of X match, send a 0 otherwise.
- (3) Send a 1 followed by Y on link l_{12} if the two copies of Y match, send a 0 otherwise.
- (4) Forward the received packet on link l_{13} .

We can show that above nonlinear error detection strategy allows a sink node to decode (X, Y) . Suppose that (a, b) and (c, d) are independent. Then coding vectors on any two links on the bottom layer are independent and they satisfy with MDS (maximum distance separable) properties. If nothing was sent down both l_{11} and l_{12} , the decoder can recover (X, Y) from the information received on links l_{10} and l_{13} . If nothing was sent down only on l_{11} , then the outputs of l_{12} and l_{13} should not be corrupted and the decoder can recover (X, Y) . Similarly, the decoder can decode correctly when nothing was sent down only on l_{12} . If all the links in the bottom layer received symbols, there is at most one erroneous link on the bottom layer, which has MDS code. Thus we can achieve rate $2 - \frac{2}{n}$ with error detection strategy.

Now we show that scalar linear network code can achieve at most rate $4/3$. Suppose that we want to achieve linear coding capacity k/n by transmitting k symbols reliably by using scalar linear network code ϕ during n time slots. To show the insufficiency of linear coding for achieving this capacity, from (1), it is sufficient to prove that there exist pairs (w, e) and (w', e') for linear network code ϕ such that

$$(\psi_l(w, e) : l \in \Gamma_+(u)) = (\psi_l(w', e') : l \in \Gamma_+(u)),$$

$N(e), N(e') \leq 1$. Since above equation is equivalent to

$$(\tilde{\phi}_l(w - w') : l \in \Gamma_+(u)) = (\theta_l(-e + e') : l \in \Gamma_+(u)),$$

by linearity, it is enough to find x and e'' such that $x \in \mathcal{X}$, $N(e'') \leq 2$, and

$$(\tilde{\phi}_l(x) : l \in \Gamma_+(u)) = (\theta_l(e'') : l \in \Gamma_+(u)), \quad (2)$$

where \mathcal{X} denotes a source alphabet and $|\mathcal{X}| = q^k$. We will show that there exists (x, e'') satisfying above equation when errors occur on the links l_1 and l_3 in error vector e'' .

Let M_1 and M_2 denote transfer matrices between a and u , and between b and u during n time slots respectively. To transmit k symbols reliably in this network, both M_1 and M_2 should have rank at least k , i.e., $\text{rank}(M_1) \geq k$ and $\text{rank}(M_2) \geq k$. Otherwise, when adversarial link is on the top layer, we cannot transmit k symbols reliably on the top layer from Theorem 1. Then data processing inequality gives us contradiction.

Let e_1 and e_2 denote the errors occurring on links l_1 and l_3 , respectively. Errors on e_1 propagates to l_{10} and l_{11} , and errors on e_2 propagates to l_{12} and l_{13} . We use $4n \times k$ matrix G_u to denote the transfer matrix between s and u during n time slots. Its columns are global coding vectors assigned on l_{10} , l_{11} , l_{12} , and l_{13} .

From (2), we have following set of equations

$$G_u x = \begin{pmatrix} M_1 & 0 \\ 0 & M_2 \end{pmatrix} (e_1, e_2)^\tau = M \cdot e''.$$

If $\text{rank}(G_u) < k$, then there exists some $x_1 \neq 0$ such that $G_u x_1 = 0$. Then $(x, e_1, e_2) = (x_1, 0, 0)$ satisfies above equation. Actually, this network code is a bad code itself since we cannot distinguish any pair of source messages w and w' such that $w - w' = x_1$ even when there are no error links in the network.

Otherwise, $\text{rank}(G_u) = k$. Since $\text{rank}(M_1) \geq k$ and $\text{rank}(M_2) \geq k$, $\text{rank}(M) \geq 2k$. Then $A = \{G_u x : x \in \mathcal{X}\}$ and $B = \{M e'' : e'' \in GF^{4n}(q)\}$ are both linear subspaces of $GF^{4n}(q)$, and $\dim(A) = k$ and $\dim(B) \geq 2k$.

Let $\{x_1, \dots, x_k\}$ denote the basis of \mathcal{X} . Then $\{G_u x_1, \dots, G_u x_k\}$ is the basis of A . Similarly, since $\text{rank}(M) \geq 2k$, there exist $2k$ vectors $\{y_1, \dots, y_{2k}\}$ such that $\{M y_1, \dots, M y_{2k}\}$ is a subset of basis of B .

If $3k > 4n$, since both A and B are linear subspaces of $GF^{4n}(q)$, there exists $(a_1, \dots, a_k, b_1, \dots, b_{2k}) \neq (0, \dots, 0)$ such that

$$\sum_{i=1}^k a_i (G_u x_i) + \sum_{j=1}^{2k} b_j (M y_j) = 0.$$

If $(a_1, \dots, a_k) = (0, \dots, 0)$ or $(b_1, \dots, b_{2k}) = (0, \dots, 0)$, then it contradicts the linear independence of basis. Thus,

$(a_1, \dots, a_k) \neq (0, \dots, 0)$ and $(b_1, \dots, b_{2k}) \neq 0$. Then,

$$\begin{aligned} & \sum_{i=1}^k a_i (G_u x_i) + \sum_{j=1}^{2k} b_j (M y_j) \\ &= \sum_{i=1}^k G_u (a_i x_i) + \sum_{j=1}^{2k} M (b_j y_j) \\ &= \sum_{i=1}^k G_u (a_i x_i) - \sum_{j=1}^{2k} M (-b_j y_j) \\ &= 0. \end{aligned}$$

Therefore, we have found nonzero $x = \sum_{i=1}^k a_i x_i$ and $(e_1, e_2)^\tau = -\sum_{j=1}^{2k} (-b_j y_j)$ such that $G_u x = M e'$. It completes the proof. \blacksquare

Corollary 1: Given a network in Fig. 1, for a single adversarial link, vector linear network code can achieve at most $4/3$.

Proof: For a network code using vector transmission, the outgoing edges of each node carries vectors of alphabet symbols which are function of the vectors carried on the incoming edges to the node. We consider a vector linear code that groups m symbols into a vector. As in Theorem 1, we define $(4n)m \times km$ generator matrix G_u between s and u . Transfer matrices M_1 and M_2 are also defined in the same way, and $\text{rank}(M_1) \geq km$ and $\text{rank}(M_2) \geq km$. As in the proof of Theorem 1, when $k > \frac{4n}{3}$, we can show that there exists vectors (x, e_1, e_2) ($x \neq 0$) satisfying

$$G_u x = (M_1 \cdot e_1, M_2 \cdot e_2).$$

C. Upper bound on the linear coding capacity

Here we present a strategy that gives an upper bound on the linear coding capacity for arbitrary network. Suppose that we want to transmit k symbols reliably by using scalar linear network code ϕ during n time slots. Then it is sufficient to find $x \in \mathcal{X}$ and e'' such that $N(e'') \leq 2z$ and

$$(\phi_l(x) : l \in \Gamma_+(u)) = (\theta_l(e'') : l \in \Gamma_+(u)). \quad (3)$$

Let $\Omega_u = \text{cut}\{\mathcal{V} - \{u\}, u\}$ denote the cut between sink node u and all other nodes. $C_1 = \sum_{l \in \Omega_u} r(l)$ denotes the volume of Ω_u . Suppose that there exists a cut Ω which contains $p \geq 2z$ links and there are m disjoint sets of links (L_1, \dots, L_m) such that $2z \geq m(p - 2z)$, $L_i \subset \Omega$, $|L_i| = p - 2z$, $L_i \cap L_j = \emptyset$, and $\Omega(L_1) \cup \dots \cup \Omega(L_m) = \Omega_u$ where $\Omega(L_i)$ denotes the set of links in Ω_u such that symbols on L_i can be propagated. We prove that $C_1/(m+1)$ is an upper bound of linear coding capacity by showing that there is (x, e'') that satisfies (3) when error vector e'' consists of error links in (L_1, \dots, L_m) .

We use e_i to denote an error vector on L_i . Let $\theta^i(e_i) = (\theta_l(e_i) : l \in \Omega(L_i))$ denote the output on $\Omega(L_i) \subseteq \Omega_u$ given e_i . Given a linear network code ϕ , let M_i denote a transfer matrix between L_i and $\Omega(L_i)$. i.e., $\theta^i(e_i) = M_i \cdot e_i$.

To transmit k symbols reliably in this network, M_i should have rank at least k , i.e., $\text{rank}(M_i) \geq k$ for $1 \leq i \leq m$. Given an error vector $e'' = (e_1, \dots, e_m)$ on the cut Ω , since $\theta_l(e) = \sum_{\{j:l \in \Omega(L_j)\}} \theta_l(e_j)$ for $l \in \Omega_u$, we obtain following equation

$$(\theta_l(e'') : l \in \Omega_u) = A \cdot (\theta^1(e_1), \dots, \theta^m(e_m))^\tau \quad (4)$$

which is equivalent to

$$\begin{aligned} & (\theta_l(e'') : l \in \Omega_u) \\ &= A \begin{pmatrix} M_1 & \dots & \dots & 0 \\ 0 & M_2 & \dots & \dots \\ \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & M_m \end{pmatrix} (e_1, \dots, e_m)^\tau \\ &= A \cdot M \cdot (e'')^\tau. \end{aligned}$$

Here a matrix A depends on the graph topology. For instance, when $L_1 = l_1$ and $L_2 = l_3$ in Fig. 3, $\Omega(L_1) \cup \Omega(L_2) = \Omega_u$ and $\Omega(L_1) \cap \Omega(L_2) = \emptyset$. Since $M \cdot (e'')^\tau = (\theta_{l_{10}}(e_1), \theta_{l_{11}}(e_1), \theta_{l_{12}}(e_2), \theta_{l_{13}}(e_2))$, and $\theta_l(e'') = \theta_l(e_1)$ for $l \in \{l_{10}, l_{11}\}$ and $\theta_l(e'') = \theta_l(e_2)$ for $l \in \{l_{12}, l_{13}\}$, $A = I_{4n}$. Since we assume that errors on (L_1, \dots, L_m) can be propagated to any link in Ω_u , i.e., $\Omega(L_1) \cup \dots \cup \Omega(L_m) = \Omega_u$, A has always full rank.

We use G_u to denote the generator matrix between s and u . Then (3) is equivalent to

$$G_u x = A \cdot M \cdot (e'')^\tau. \quad (5)$$

Since $\text{rank}(M) = \sum_{i=1}^m \text{rank}(M_i) \geq km$ and A has full rank, $\text{rank}(AM) = \text{rank}(M) \geq km$.

If $\text{rank}(G_u) < k$, then there exists some $x_1 \neq 0$ such that $G_u x_1 = 0$. Then $(x, e'') = (x_1, 0)$ satisfies (4) and this network code is a bad code itself.

When $\text{rank}(G_u) = k$, since $\text{rank}(M) \geq mk$, we can always find (x, e_1, e_2) satisfying (4) when $k + mk > C_1$. Thus, the upper bound on the achievable linear coding capacity is $C_1/(m+1)$.

V. LINEAR AND NONLINEAR OPERATIONS AT INTERMEDIATE NODES

In the unit link capacities problem, coding only at the source and simple forwarding at intermediate nodes suffices for a single source and single sink. Here we give examples where intermediate nodes may have to do coding or error correction for achieving the capacity in our unequal link capacities problem. We have already shown that error detection at intermediate nodes can be used for achieving capacity. We conjecture that the generalized Singleton bound is not achievable in general.

A. coding at intermediate nodes

Here we give an example network that coding at intermediate nodes but not error-detection and correction is necessary for achieving the capacity. We consider a 3-hop single source and single sink network in Fig. 4. For a single adversarial link, $\min_{\Omega \in CS(s,u)} M(\Omega) = 4$.

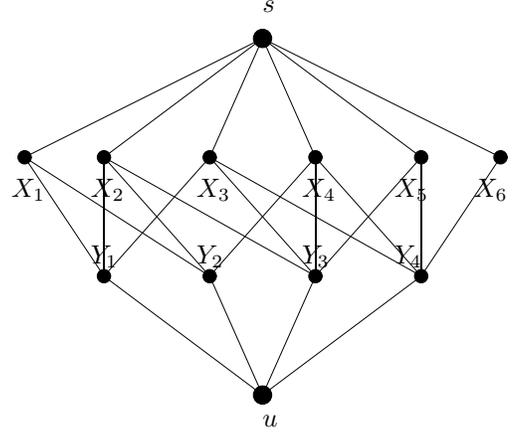


Fig. 4. All links on the top or middle layer have capacity one. All links on the bottom layer have capacity 2.

Lemma 3: Given a network in Fig. 4, for a single adversarial link, coding at intermediate nodes is necessary for achieving the rate 4.

Proof: To achieve rate 4, any four links on the top layer should carry 4 independent packets. Otherwise, when adversarial link is on the top layer, source cannot transmit 4 packets reliably Theorem 1. Then data processing inequality gives us contradiction. Similarly, any two links on the bottom layer should carry 4 independent packets. Since Y_i is connected to at most four different nodes among (X_1, \dots, X_6) for $\forall 1 \leq i \leq 4$ and all links in the middle layer have capacity 1, each of $Y_1, Y_2, Y_3,$ and Y_4 receives all independent information. Thus we cannot apply error-detection or correction at $Y_1, Y_2, Y_3,$ and Y_4 . Suppose that only forwarding strategy is used on this network. Then we show that rate 4 is not achievable. There are six symbols on the top layer. Since we use only forwarding, these are forwarded to the bottom layer. Since bottom layer links have total capacity 8, there are at least two same symbols on the bottom layer links. This contradicts that any two links on the bottom layer should carry four independent information to achieve rate 4. Therefore forwarding is insufficient for achieving the rate 4 in this network.

Now we show that a generic linear network code, where intermediate nodes do coding achieve rate 4. From [6, Ch 19], generic network code can be constructed with high probability by randomly choosing the global encoding kernels provided that the base field is much larger than sufficient. So when we apply random linear network code on this network, it is generic with high probability when q is very large. If adversarial link is on the top or middle layer, then each capacity 2 on the bottom layer is equivalent to two unit capacity links. Then all links in the network have capacity one and this problem is reduced to the equal link capacities problem. From [2], rate $6 - 2 \times 1 = 4$ is achievable. From [6,

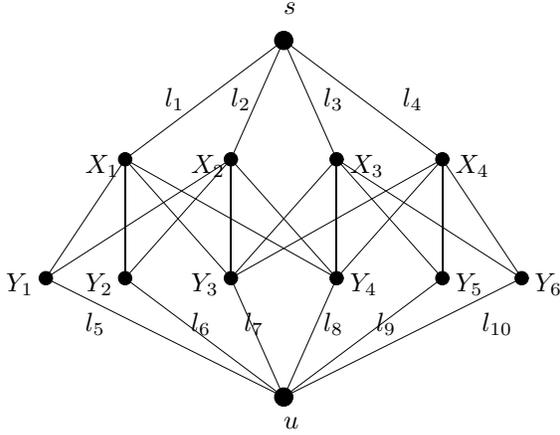


Fig. 5. The link capacity in this network is as follows: $r(l_1) = r(l_2) = r(l_3) = r(l_4) = 4, r(l_5) = \dots = r(l_{10}) = 2$. All the links in the middle layer have capacity 1.

Theorem 19.32], since the min-cut between s and (Y_i, Y_j) is at least 4 for $\forall 1 \leq i \neq j \leq 4$, in a generic network code the global encoding kernels on any two links on the bottom layer are linearly independent and they satisfy with MDS property. Thus an error on the last layer can be corrected. ■

B. error correction at intermediate nodes

In this section, we give an example in which error correction at intermediate nodes is used for achieving the capacity. The intuition behind our approach is that error correction at intermediate nodes can reduce the error propagation to the bottom layer and MDS code assigned on the bottom layer gives the correct output. We consider a 3-hop single source-destination network in Fig. 5. For a single adversarial link, $\min_{\Omega \in CS(s,u)} M(\Omega) = 8$. From Sec. III-C, when $\Omega = \{l_1, l_2, l_3, l_4\}$, $L_1 = \{l_1, l_4\}$, and $m = 1$, the upper bound on the linear coding capacity is $\sum_{i=5}^{10} r(l_i)/(m+1) = 6$.

Lemma 4: Given a network in Fig. 5, for a single adversarial link, rate 8 is achievable using error correction at intermediate nodes.

Proof: Without loss of generality, all nodes forward the received information except Y_3 and Y_4 . We first assign (12, 8) MDS code (a, b, \dots, l) on the bottom layer links and apply (4, 2) MDS code at each decision node, e.g., assign $(e, f, e+f, e+2f)$ and $(g, h, g+h, g+2h)$ on incoming links to Y_3 and Y_4 respectively. Then we can assign codewords on all links in the network since all nodes except Y_3 and Y_4 are forwarding. If adversarial link is on the middle or bottom layer, at most two errors are propagated to the sink node and MDS code assigned on the bottom layer gives the correct output. If adversarial link is on the top layer, at most two errors are propagated to the sink node through forwarding nodes Y_1, Y_2, Y_5 , and Y_6 . Since at most one error is incoming to Y_3 and Y_4 respectively, (4, 2) MDS code applied at each decision node gives error-free output (e, f)

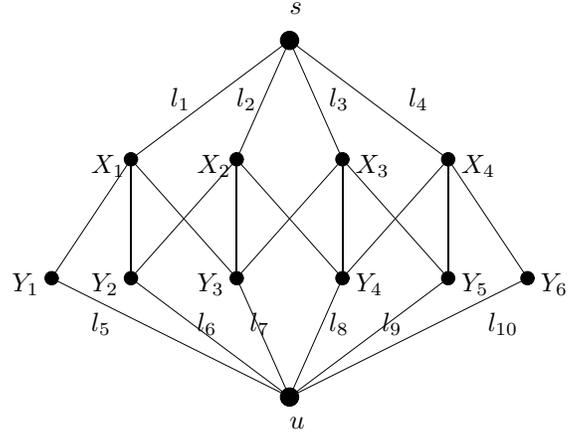


Fig. 6. The link capacity in this network is as follows: $r(l_1) = r(l_2) = r(l_3) = r(l_4) = 3, r(l_6) = r(l_7) = r(l_8) = r(l_9) = 2, r(l_5) = r(l_{10}) = 1$. All links in the middle layer have capacity 1.

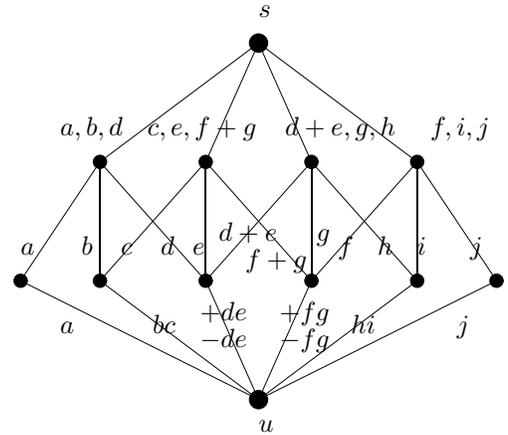


Fig. 7. At nodes Y_3 and Y_4 , we apply error-detection. At Y_3 , if any two of $(d, e, d+e)$ gives the same output, it transmits $+(d, e)$. Otherwise, it transmits $-(d, e)$ where d and e are inputs from X_1 and X_2 and they can be corrupted. We do the same operation at node Y_4 .

and (g, h) . Therefore, when adversarial link is on the top layer, at most two errors are propagated to the sink and (12, 8) MDS code returns the correct output. ■

C. conjecture

Conjecture 1: Generalized Singleton bound is not achievable in general.

We believe the example in Fig. 6 is one in which the generalized Singleton outer bound is not achievable. For a single adversarial link, $\min_{\Omega \in cut(s,u)} M(\Omega) = 6$. From the test in Sec. III-C, when $\Omega = \{l_1, l_2, l_3, l_4\}$, $L_1 = \{l_1, l_4\}$, and $m = 1$, the the upper bound on the linear coding

capacity is $\sum_{i=5}^{10} r(l_i)/(m+1) = 5$. Now we consider nonlinear error detection strategy with MDS code and show that this is insufficient for achieving rate 6. (a, b, \dots, j) forms $(10, 6)$ MDS code. As shown in Fig. 7, we assign codewords on each link in the top layer and middle layer based on MDS code. Without loss of generality, all nodes except Y_3 and Y_4 are forwarding. We assign $(d, e, d+e)$ and $(f, g, f+g)$ on incoming links to Y_3 and Y_4 respectively. At Y_3 , if any two of $(d, e, d+e)$ gives the same output, it transmits $+(d, e)$. Otherwise, it transmits $-(d, e)$ where d and e are inputs from X_1 and X_2 respectively and they can be corrupted. We do similar operation at node Y_4 . If u receives $-(d, e)$, one of l_1, l_2, l_3 can be adversarial link. Suppose that source transmits a message w_1 . If l_1 is an adversarial link, (c, e, f, g, h, i, j) are not corrupted at the sink. In this case, by changing (a, b, d) to (a', b', d') on l_1 , it is able to make that any 6 of $(a', b', c, d', e, f, i, j)$ returns the message w_2 different from w_1 . Similarly, if source transmits w_2 and l_3 is an adversarial link, (a, b, c, d, e, f, i, j) are not corrupted. By changing (g, h) to (g'', h'') on l_3 , we can make any 6 of $(c, e, f, g'', h'', i, j)$ returns w_1 . Then, we construct two confusable sequences at the sink and cannot distinguish w_1 from w_2 . Therefore, error detection strategy described above cannot be used for achieving rate 6.

VI. CONCLUSION AND FUTURE WORK

In this paper, we propose a new problem, network error correction with unequal link capacities. We define z -error link-correcting code that can correct all errors occurring up to z links in the network. We have found the capacity of two-node network with parallel links and extend this result to the generalized Singleton outer bound that can be applied for all network scenarios. We have shown by example that linear coding is insufficient for achieving capacity in general and presented a method for finding an upper bound on the linear coding capacity for arbitrary network. We show that unlike the equal link capacity case, even for a single source and single sink network, it may be necessary for intermediate nodes to do coding, nonlinear error detection or error correction. We conjecture that the generalized Singleton bound is not achievable in general, even using nonlinear operations. Future work includes to prove or disprove this conjecture, to investigate the error correction capacity of general network and to develop low complexity achievable schemes.

ACKNOWLEDGEMENTS

This work is partly supported under a subcontract #069153 issued by BAE Systems National Security Solutions, Inc. and supported by the Defense Advanced Research Projects Agency (DARPA) and the Space and Naval Warfare System Center (SPAWARSYSCEN), San Diego under Contract No. N66001-08-C-2013. We thank Nitin Vaidya for pointing out an error in a previous version of this paper.

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] R. W. Yeung and N. Cai, Network error correction, Part II: Lower bounds, *Communications in Information and Systems*, vol. 6, no. 1, pp. 19-36, 2006.
- [3] Z. Zhang, "Linear Network Error Correction Coding in Packet Networks," *IEEE Tran. on Inform. Theory*, vol. 54, no. 1, pp. 209-218, Jan. 2008.
- [4] S. Yang, C. K. Ngai, and R. W. Yeung, "Construction of linear network codes that achieve a refined singleton bound," in *Proc. ISIT*, June 2007.
- [5] S. Yang, and R. W. Yeung, "Refined coding bounds for network error correction," in *Proc. ITW*, July 2007.
- [6] R. W. Yeung, *Information theory and network coding*, Springer, 2008.