Exact Modeling of the Performance of Random Linear Network Coding in Finite-buffer Networks

Nima Torabkhani†, Badri N. Vellambi‡, Ahmad Beirami†, Faramarz Fekri†
† School of Electrical and Computer Engineering, Georgia Institute of Technology
‡ Institute for Telecommunications Research, University of South Australia
E-mail: {nima, beirami, fekri}@ece.gatech.edu, badri.vellambi@unisa.edu.au

***Abstract*—In this paper, we present an exact model for the analysis of the performance of Random Linear Network Coding (RLNC) in wired erasure networks with finite buffers. In such networks, packets are delayed due to either random link erasures or blocking by full buffers. We assert that because of RLNC, the content of buffers have dependencies which cannot be captured directly using the classical queueing theoretical models. We model the performance of the network using Markov chains by a careful derivation of the buffer occupancy states and their transition rules. We verify by simulations that the proposed framework results in an accurate measure of the network throughput offered by RLNC. Further, we introduce a class of acyclic networks for which the number of state variables is significantly reduced.**

I. Introduction

It is well-known that linear network codes achieve the min-cut capacity of networks for unicast applications [1]. In fact, random linear codes over large Galois fields suffice to achieve the min-cut capacity [2]. *Random linear network coding* (RLNC) has been shown to improve the performance in distributed settings with time-varying network parameters. In these networks, a distributed and packetized network coding scheme, where each node stores received packets and forwards random linear combinations of the stored packets when required, was introduced in [3]. As a result, for a network of nodes with no buffer limitations, all arriving packets at a node are stored, and then used to generate new packets to send. Hence, there is no information loss. However, in this case, upon reception of a packet, a node has to determine whether or not the incoming packet is in the linear span of its previously stored packets. Further, for generating every coded packet, all stored packets need to be accessed. It is therefore desirable to have limited buffer sizes, since it limits the complexity of storage and coded packet generation process. Further, using small buffers at relay nodes simplifies practical issues such as on-chip board space and memory-access latency as well as reducing the average packet delay [4], [5].

The problem of computing capacity and designing efficient coding schemes for erasure networks has been widely studied in the absence of buffer constraints [1], [6], [7]. The limitations posed by finite buffers were considered by [8], specifically in a simple two-hop line network. Inspired by this work, in [9], the authors present a Markov-chain-based approach to model the dynamics of the system and the packet occupancy of every intermediate node to approximate the performance parameters (throughput and latency) of a multi-hop line network with lossy links. Several challenges arise when extending the study from a single intermediate node to a multi-hop line network. Results from [9] were extended to other communication scenarios, such as block-based random linear coding for line networks [10], and general wired networks with lossless feedback and random routing [11]. However, the main challenge of modeling the evolution of *buffer occupancy* or *innovativeness of buffer contents* in general network topologies when RLNC is used, was not addressed in these works.

The queueing theory framework for lossy networks with finite buffers of [12], [13] attempts to model the packets of the network as customers, the delay due to packet loss over links as service times in the nodes, and the buffer size at intermediate nodes as the maximum queue size. However, this packet-customer equivalence fails to accurately model RLNC in general network topologies. This is due to the possibility of packet replication at intermediate nodes, or more generally, the potential correlation in the contents of the buffers of various intermediate nodes. This correlation or dependency between contents of the buffers cannot be captured directly in the customer-server based queueing model.

In this paper, our objective is to study the relation between throughput of RLNC and the buffer sizes of intermediate nodes in the small buffer regime. The first and the key step in our approach is to derive using algebraic tools the state of the buffers using which the dynamics of the network can be completely characterized. We then derive the state update rules for each transmission in the network. Finally, using the developed state space and update rules, we obtain the throughput of the network using Monte Carlo simulations and compare the results to the actual packetized implementation of RLNC. We believe the proposed modeling framework is a significant step towards developing a theoretical framework for computing the throughput capacity and the packet delay distribution in general finite-buffer wired networks.

This paper is organized as follows. First, we present a formal definition of the problem and the challenges in Section II. Next, we investigate the tools and steps for modeling the buffer states in Section III. We then introduce in Section IV, a general class of networks for which the complexity of our modeling is significantly lesser. Finally, Section V presents our model validation results using simulations. Conclusions are summarized in Section VI.

II. Problem Setup and Challenges

Throughout this work, we model the network by an acyclic directed graph $\overrightarrow{G}(V, \overrightarrow{E})$, where packets can be transmitted over a link $\overrightarrow{e} = (u, v)$ only from the node u to v. The system is analyzed using a discrete-time model; each node can

This material is based upon work supported by the National Science Foundation under Grant No. CCF-0914630, and the Australian Research Council under ARC Discovery Grants DP0880223 and DP1094571.

transmit at most one packet over a link in an epoch. The loss process on each link is assumed to be memoryless, i.e., packets transmitted on a link $\overrightarrow{e} = (u, v) \in \overrightarrow{E}$ are lost randomly with a probability of $\varepsilon_{\overrightarrow{e}} = \varepsilon_{(u,v)}$. Note that the erasures are due to the quality of links (e.g., noise, interference) and do not represent packet blocking due to finiteness of the buffers. Further, the packet loss processes on different links are assumed to be independent. Each node $v \in V$ has a buffer size of m_v packets with each packet having a fixed size. Source and destination are assumed to be able to store an infinitude of packets. Throughout this paper, node s and node d represent the source and destination nodes, resp. Also, for any $x \in [0, 1]$, $\overline{x} \triangleq 1 - x$. The unicast information-theoretic throughput is also defined as the expected rate (in packets/epoch) at which information packets are transferred from the source to the destination when the network is in steady-state. In other words, if τ_k is the time it takes for k information packets to be transmitted to the destination, the throughput capacity is given by

$$\mathcal{C}(\overrightarrow{G}) = \lim_{k\to\infty} (\tau_k)^{-1} k. \tag{1}$$

There are two key challenges in finite-buffer networks. The first challenge is the choice of optimal buffer management strategy, which also depends on the routing/coding scheme that is in use. Due to losses on links, and finiteness of buffers, transmission of a packet by a node u on $\overrightarrow{e} = (u, v)$ does not guarantee successful reception by the node v. Thus, in the absence of any feedback, a node u does not know if it can delete a packet from its buffer to make room for its next incoming packet. Further, it is also unclear if transmitting a packet via several parallel paths will increase the throughput. The second challenge is due to the possible replication of packets in the network. Hence, it is neither possible to model the system dynamics by a simple queueing model where packets are customers and the buffers as queue sizes, nor is it feasible to treat the packets as flows in the network.

Random Linear Network Coding (RLNC) attractively bypasses these two challenges. It eliminates the need for a feedback strategy to delete stored packets because the physical act of storing a packet becomes immaterial. It also eliminates the need for active replication by allowing transmitted/stored packets to be treated as elements of an abstract vector space. This makes RLNC a favorable choice for practical schemes in finite-buffer scenarios.

We consider the following packet-coding scheme introduced in [8], which is a finite-buffer adaptation of RLNC. In this scheme, at each epoch, random linear coding is used for both the packet generation and storage by intermediate nodes. As an example, consider a node u of buffer size m_u. At a given epoch, u generates an encoded packet by performing a random linear combinations of m_u stored data packets (over a sufficiently large Galois field[1] $\mathbb{F}_q$), and transmits the coded packet on an outgoing link. For storage, when a packet successfully arrives at a node v, the node multiplies the received packet by a random vector chosen uniformly from $\mathbb{F}_q^{m_v}$, and adds the resultant vector components to each of the present buffer contents.

Therefore, using RLNC, after just a single packet reception, the entire buffer becomes physically full with multiples of the received packet. Thus, even though the buffer of the node u is almost always physically full, the number of stored packets that is innovative w.r.t any other subset of nodes can vary from 0 to m_u. As an example, suppose that two nodes a and b receive/store two packets each generated from three original packets from a relay c. In this case, a and b will have two innovative packets each for the destination. Now, suppose a delivers a packet to the destination. Then, b still contains two innovative packets for the destination. However, if a delivers another packet to the destination, b will only have one innovative packet for the destination, since both nodes together originally possessed only three innovative packets for the destination. In this example, the challenges of tracking the number of innovative packets and the interdependency between buffer contents gets compounded further as the packets from a and b are propagated to the other intermediate nodes. This interdependency between buffer contents signals the need for a novel notion of *occupancy* to track the number of innovative packets each node has for the destination, and consequently, to determine the throughput capacity of the network. This notion will be formalized in the following section.

The main motivating factor to develop a theoretical model for these networks is to understand the throughput capacity under RLNC. In order to measure the throughput of RLNC in these networks, one option is to perform a Monte Carlo simulation where encoded packets are generated using coefficients in a large finite field $\mathbb{F}_q$, and buffer updates are performed upon each successful reception. This is a significantly time-consuming simulation due to large field operations. A theoretical model that tracks buffer dynamics based on occupancy of buffers will be a simpler alternate means. As we will see, the developed model provides a more efficient way of measuring the performance of finite-buffer networks. Additionally, it provides us with intuitive insights on the dynamics of buffer updates, which is a major step towards computing performance metrics for such networks, and analyzing their key trade-offs.

III. Exact Modeling of Finite-buffer RLNC

Here, we introduce the tools and steps that enable us to track changes in the buffer contents of nodes.

To identify the throughput as defined in (1), we assume that the source possesses a sufficiently large block of packets that has to be transmitted to the destination. The first aim is to formalize the notion of buffer occupancy by investigating the dimension of the span of the stored packets in the buffers. Let $\{T_1, T_2, \ldots, T_k\}$ be the original information packets at the source. Let $[n] \triangleq \{1, 2, \ldots, n\}$ denote the set of all intermediate nodes, where $n = |V| - 2$. Let $P_{i,j}(t)$ be the packet contained in buffer slot j of relay i at time epoch t, where $P_{i,j}(t) = \sum_{l=1}^{k} a_{i,j,l} T_l$, $i \in [n]$, $j \in [m_i]$, and $a_{i,j,l}$ is a coefficient in the chosen Galois field $\mathbb{F}_q$. Let $\mathcal{V}(S)(t) \triangleq \text{span}\{P_{i,j}(t) |\ j \in [m_i], i \in S\}$ for all $S \subseteq [n]$. To simplify the notations, we will drop the reference to time in $\mathcal{V}(S)(t)$ by using $\mathcal{V}(S)$. Also, we define $S^c \triangleq [n] \setminus S$.

Definition 1: For any two subsets of the intermediate nodes $S, S' \subseteq [n]$, we define the *innovativeness* of S w.r.t. S' at time instant t as:

$$I_{S\to S'} = \dim\big(\mathcal{V}(S)\big) - \dim\big(\mathcal{V}(S) \cap \mathcal{V}(S')\big). \tag{2}$$

[1]The size of the Galois field needs to be sufficiently large to increase the chance of innovativeness of the coded packet.

In other words, $I_{S\to S'}$ gives the number of innovative packets that buffer contents of nodes in S can generate which cannot be generated by the contents of the buffers of nodes in S'.

Definition 2: The occupancy vector $\{b_S\}_{S\subseteq[n]}$ of the network is defined [2] to be

$$b_S \triangleq \dim\big(\mathcal{V}(S)\big) - \dim\big(\mathcal{V}(S)\cap\mathcal{V}(S^c)\big),\ S\subseteq[n]. \quad (3)$$

The following lemma shows that the knowledge of occupancy vector $\{b_S\}_{S\subseteq[n]}$ is equivalent to knowing the innovativeness of any subset of the relay nodes w.r.t. any other subset. This result significantly reduces the number of state space variables.

Lemma 1: For $S, S'\subseteq[n]$, $I_{S\to S'} = b_{S'^c} - b_{\{S\cup S'\}^c}$.

Proof 1: Proof omitted due to lack of space.

Since the occupancy vector provides the innovativeness of the contents of each node w.r.t the remaining nodes, we need to be able to track the dynamics of the occupancy vector for successful transmissions on links to complete the system modeling. To do so, let superscripts $-$ and $+$ denote the status of a system parameter before and after a successful packet transmission on a link. The following results derive the rules for updating the occupancy vector when successful transmissions occur. Throughout these results, we denote *whp*/*wlp* to qualify an event if its probability of occurrence can be made arbitrarily close to unity/zero by increasing the field size alone.

Lemma 2: (*Source-to-Relay*) The update rules when a relay i successfully receives a packet from s are as follows *whp*.

- If $i\in S\subseteq[n]$ and $b_{\{i\}} < m_i$, then $b_S^+ = b_S^- + 1$.
- If $i\notin S\subseteq[n]$, $b_{\{i\}} < m_i$ and $I^-_{\{i\}\to S^c\setminus\{i\}} = m_i$, then $b_S^+ = b_S^- + 1$.
- Otherwise, $b_S^+ = b_S^-$.

Proof 2: Proof omitted due to lack of space.

Lemma 3: (*Relay-to-Relay*) The update rules when relay j successfully receives a packet from relay i are as follows *whp*.

- If $i\in S\subseteq[n]$, $j\in S^c$, $I^-_{\{j\}\to S^c\setminus\{j\}} < m_j$ and $I^-_{\{i\}\to S^c} > 0$, then $b_S^+ = b_S^- - 1$.
- Otherwise, $b_S^+ = b_S^-$.

Proof 3: See Appendix A.

Lemma 4: (*Relay-to-Destination*) The update rules when d successfully receives a packet from relay j are as follows *whp*.

- If $i\in S\subseteq[n]$ and $I^-_{\{i\}\to S^c} > 0$, then $b_S^+ = b_S^- - 1$.
- Otherwise, $b_S^+ = b_S^-$.

Proof 4: Proof omitted due to lack of space.

On the whole, an update of buffer occupancy occurs only when the delivered packet is innovative for the receiving node and the buffer of the receiving node is not full. Next, we describe how the state update rules could be utilized to obtain the throughput of a network. Let $\overrightarrow{E}^* = (\overrightarrow{e}_1,\ldots,\overrightarrow{e}_{|\overrightarrow{E}|})$ be an ordering of the edge set $\overrightarrow{E}$, and let $l(t)\in\{0,1\}^{|\overrightarrow{E}|}$ represent the realization of the channels at time t. That is $l_i(t)=1$ if the i^{th} edge $\overrightarrow{e}_i$ in $\overrightarrow{E}^*$ does not erase the transmitted packet during the epoch t. Then, given the occupancy vector $\{b_S(t)\}_{S\subseteq[n]}$ and the channel realization $l(t)$, the occupancy vector $\{b_S(t+1)\}_{S\subseteq[n]}$ can be determined using the state update rules presented in Lemmas 2, 3, 4.

Further, the state transition probability matrix $\mathbb{T}$ for the corresponding Markov chain can be identified as follows. Also, let $T_{\overrightarrow{e}}$ be the state transition matrix given a successful packet transmission on the link $\overrightarrow{e}$. For any $\overrightarrow{e}\in\overrightarrow{E}$, $T_{\overrightarrow{e}}$ can be determined using Lemmas 2, 3, 4. Therefore,

$$\mathbb{T} = \sum_{l\in\{0,1\}^{|\overrightarrow{E}|}} \Big(\prod_{j:l_j=0}\varepsilon_{\overrightarrow{e}_j}\Big)\Big(\prod_{i:l_i=1}\overline{\varepsilon}_{\overrightarrow{e}_i}T_{\overrightarrow{e}_i}\Big). \quad (4)$$

This Markov chain can be proved to be *irreducible*, *aperiodic*, and *ergodic* [9], [14]. Therefore, it possesses a unique steady-state probability distribution. Moreover, due to ergodicity, the time averages are equivalent to the statistical averages. Therefore, the throughput capacity $\mathcal{C}(\overrightarrow{G})$ can be determined using the steady state probability of the event that the network is in a state wherein the nodes possessing a link to the destination have innovative packets as follows.

$$\mathcal{C}(\overrightarrow{G}) = \sum_{l\in\{0,1\}^{|\overrightarrow{E}|},\{b_S(t)\}} \mathfrak{N}(l,\{b_S(t)\})\cdot\Pr\Big(\{b_S(t)\}\Big), \quad (5)$$

where $\mathfrak{N}(l,\{b_S(t)\})$ represents the number of successfully transmitted packets when state $\{b_S(t)\}$ and channel realization l occur together.

IV. State Size Reduction in a Class of Networks

In Section III, we observed that the number of state variables that we need to track at each time epoch is 2^n-1 since b_S, the innovativeness of every subset of relay nodes w.r.t. its complement, must be considered. In this section, we show that all innovativeness terms need not be tracked to completely define the state of the system. This is a consequence of the intuition gained in line networks [9]. In line networks, we need to only track $I_{i\to S}$, where $S=\{i+1,\ldots,n\}$, i.e., all those intermediate nodes that are farther from the source hop-distance-wise. Equivalently, for line networks, it suffices that we track b_S for $S=\{1,\cdots,i\}$ for $i\in[n]$. Extending that intuition, define $\mathscr{A}\triangleq\{S\subseteq[n]:$ Every $j\in S^c$ has a path in S^c to $d\}$ as illustrated in Fig. 2. Consider a partition of the set of relay

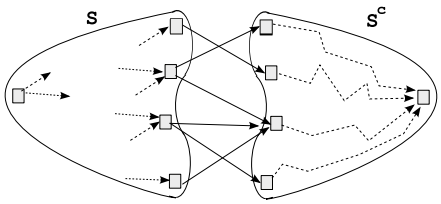


Fig. 1. Illustration of a set S in $\mathscr{A}$.

nodes into types $\{H_1,H_2,\ldots\}$, where a relay node v belongs to H_k if the shortest hop-distance from v to the destination d is k, and $H_0\triangleq\{d\}$. Define a class of networks $\mathcal{N}$ where every link starts at some node in H_i for some i and ends at some node in H_{i-1}. Figure 2 illustrates a network from this class. This structure enables us to track significantly lesser number of innovativeness components using the following result, which shows that tracking the occupancy for sets in $\mathscr{A}$ suffices to define the system completely.

[2]The precise definition of the occupancy vector must consider the packets that have already reached $\{d\}$ by using $b_S\triangleq\dim(\mathcal{V}(S))-\dim(\mathcal{V}(S)\cap\mathcal{V}(S^c\cup\{d\}))$. However, the inclusion of $\{d\}$ affects update rules only when dealing with the destination. For simplicity, the equivalent definition without the inclusion of $\{d\}$ is used in all cases not involving the destination.

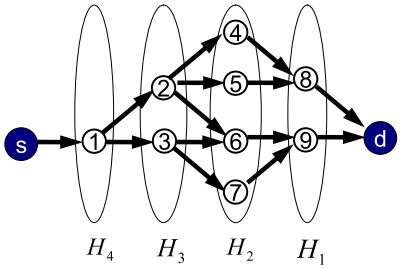


Fig. 2. An example of a directed acyclic network in $\mathcal{N}$.

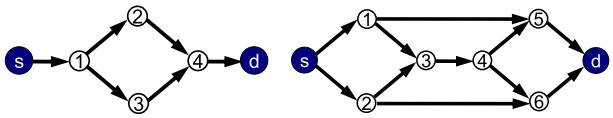
Theorem 1: For any directed acyclic network in $\mathcal{N}$, we need only track b_S for $S \in \mathscr{A}$.

Proof: Proof omitted due to lack of space. ■

V. Simulation Results

In this section, we present the results of our performance modeling framework using state update rules in comparison with an actual packetized implementation of RLNC, and will show that our framework accurately models the buffer dynamics of the network.

We consider Network 1 and Network 2 shown in Fig. 3 to compare the results of our simulations. In Network 1, the



(a) Network 1. (b) Network 2.

Fig. 3. Networks Considered for simulation

edges have erasure probabilities $\varepsilon_{(s,1)} = 0.1$, $\varepsilon_{(1,2)} = 0.6$, $\varepsilon_{(1,3)} = 0.5$, $\varepsilon_{(2,4)} = 0.4$, $\varepsilon_{(3,4)} = 0.5$, and $\varepsilon_{(4,d)} = 0.1$. In Network 2, all the edges have $\varepsilon = 0.5$ except the edges $\{(s,1),(s,2),(5,d),(6,d)\}$ for which $\varepsilon = 0.25$. All the intermediate nodes are assumed to have the same buffer size. In order to measure the exact performance parameters of this network, a block of size $k = 10^5$ packets is sent from the source to the destination. Fig. 4 and Fig. 5 present the

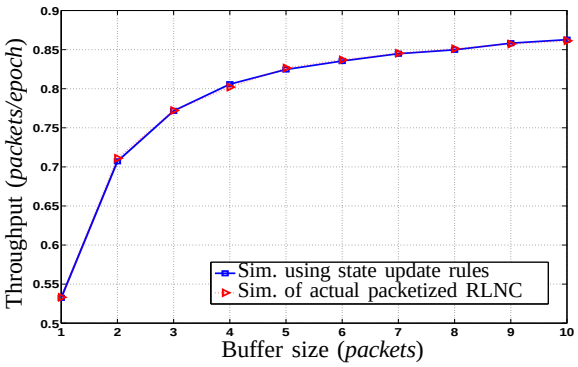


Fig. 4. Throughput of Network 1 for different buffer sizes.

variations of the throughput measured by actual simulation of RLNC and the throughput measured by simulation based on the state update rules developed in our work versus the buffer size. As it can be observed, our model is very close to the actual simulation results. Further, it confirms the optimality of RLNC for the infinite buffer setting as the curve approaches to the min-cut capacity for both networks. It is notable that

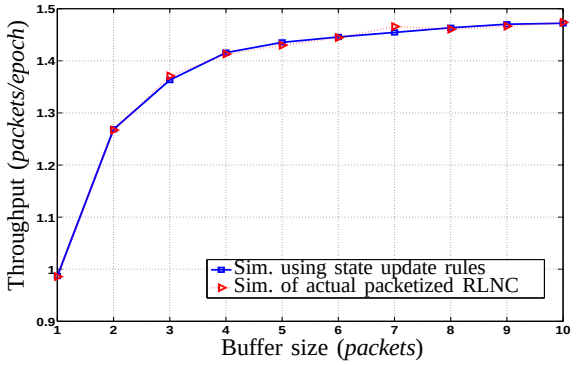


Fig. 5. Throughput of Network 2 for different buffer sizes.

the emulation of the RLNC using the derived state update rules takes significantly lesser time than the exact simulation of the RLNC scheme. Table I compares the number of states

TABLE I
Number of active states vs. buffer size in Network 1.

Buffer Size	No. of Active States	Upper Bound $(m+1)^{15}$
1	44	32768
2	600	14348907
3	4358	1073741824

actually visited (identified by simulations) and a crude upper bound on the number of states in the Markov chain model. For Network 1, the number of state variables is $2^4 - 1 = 15$, and a provable upper bound for the number of states is $(m+1)^{15}$, where m is the buffer size of each intermediate node. However, it is noticed from simulations that the number of states that is actually realized is much lesser than the bound. This observation signals suggests that a closer look at the Markov chain to reduce its size can simplify the model, thereby rendering it more easily tractable.

VI. Conclusion and Future Work

We have derived a novel notion of buffer occupancy for RLNC in wired finite-buffer networks. Using this notion, we developed a Markov-chain-based framework that can identify the throughput offered by RLNC using Monte Carlo simulations. This framework offers significant computational benefits over a complete simulation of RLNC. Though the size of the Markov chain is exponential, simulations suggest that a very small portion of the state space is actually visited in reality. A closer look at the state space and a thorough analysis to reduce the state space needs to be performed to eventually derive analytical throughput estimates.

References

[1] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Information Theory*, vol. 49, pp. 371–381, Feb. 2003.

[2] R. Koetter and M. Medard, "An algebraic approach to network coding," *IEEE/ACM Trans. on Networking*, vol. 11, pp. 782–795, Oct. 2003.

[3] P. Chou, Y. Wu, and K. Jain, "Practical network coding," *in 41st Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL*, Oct. 2003.

[4] G. Appenzeller, I. Keslassy, and N. McKeown, "Sizing router buffers," *SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 4, pp. 281–292, 2004.

[5] M. Enachescu, Y. Ganjali, A. Goel, N. Mckeown, and T. Roughgarden, "Routers with very small buffers," in *IEEE INFOCOM*, 2006.

[6] A. F. Dana, R. Gowaikar, R. Palanki, B. Hassibi, and M. Effros, "Capacity of wireless erasure networks.," *IEEE Trans. Information Theory*, vol. 52, no. 3, pp. 789–804, 2006.
[7] P. Pakzad, C. Fragouli, and A. Shokrollahi, "Coding schemes for line networks," *in IEEE Intl. Symposium on Information Theory*, Sept. 2005.
[8] D. S. Lun, P. Pakzad, C. Fragouli, M. Medard, and R. Koetter, "An analysis of finite-memory random linear coding on packet streams," *in the 2nd Workshop on Network Coding, Theory, and Applications (NetCod 2006), Boston, MA, April 3-7, 2006*.
[9] B. N. Vellambi, N. Torabkhani, and F. Fekri, "Throughput and latency in finite buffer line networks," *IEEE Trans. Information Theory*, vol. 57, pp. 3622–3643, June 2011.
[10] N. Torabkhani, B. N. Vellambi, and F. Fekri, "Study of throughput and latency in finite-buffer coded networks," *in 44th Asilomar Conference on Signals, Systems and Computers*, Nov. 2010.
[11] N. Torabkhani, B. N. Vellambi, and F. Fekri, "Throughput and latency of acyclic erasure networks with feedback in a finite buffer regime," *in IEEE Information Theory Workshop (ITW), Dublin, Ireland*, Aug. 2010.
[12] T. Altiok, "Approximate analysis of queues in series with phase-type service times and blocking," *Operations Research*, vol. 37, pp. 301–310, July 1989.
[13] T. Altiok, "Approximate analysis of exponential tandem queues with blocking," *European Journal of Operational Research*, vol. 11, no. 4, pp. 390–398, 1982.
[14] W. Feller, "An introduction to probability theory and its applications," *John Wiley & Sons, 2nd ed., 1957*.

APPENDIX A
PROOF OF LEMMA 3

From Definition 2 it is clear that if $i,j \in S$, then $b_S^+ = b_S^-$. The same applies when $i,j \in S^c$. For the case $i \in S^c, j \in S$, the update rule is $b_S^+ = b_S^-$ and the proof is similar to the one presented for the case $i \in S, j \in S^c$, which is as follows.

Hence, here we only assume $i \in S, j \in S^c$. Let $\mathcal{A}^- = \{A_1^-, A_2^-, \ldots, A_{m_i}^-\}$, $\mathcal{B}^- = \{B_1^-, B_2^-, \ldots, B_{|\mathcal{B}^-|}^-\}$, $\mathcal{C}^- = \{C_1^-, C_2^-, \ldots, C_{m_j}^-\}$ and $\mathcal{D}^- = \{D_1^-, D_2^-, \ldots, D_{|\mathcal{D}^-|}^-\}$ be the buffer contents of relay i, relays $S \setminus \{i\}$, relay j, and relays $S^c \setminus \{j\}$ before packet transmission, respectively. Suppose packet $E = \sum_{l=1}^{m_i} \alpha_l A_l^-$ successfully transfers from relay i to relay j. Then, for any $S \subseteq [n]$, We will have $\mathcal{A}^+ = \mathcal{A}^-$, $\mathcal{B}^+ = \mathcal{B}^-$, $\mathcal{D}^+ = \mathcal{D}^-$, and $\mathcal{C}^+ = \{C_1^- + \beta_1 E, C_2^- + \beta_2 E, \ldots, C_{m_j}^- + \beta_{m_j} E\}$. Note that the coefficients α_l and β_k are chosen randomly from $\mathbb{F}_q$. Let $\mathcal{G}^- = \text{span}\{\mathcal{A}^-\} \cap \text{span}\{\mathcal{C}^- \cup \mathcal{D}^-\}$. We consider two cases:

- **Case 1**: Suppose there exists λ_l, θ_k such that $\lambda_l \neq 0$ for at least one l and $\sum_l \lambda_l C_l^- + \sum_k \theta_k D_k^- = 0$. Hence,

  $$\sum_l \lambda_l C_l^+ + \sum_k \theta_k D_k^- = (\sum_l \lambda_l \beta_l) E \in \quad \text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\}$$

  Therefore, $E \in \text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\}$ whp. Further, if $\mathcal{G}^- \neq \text{span}\{\mathcal{A}^-\}$, then $E \notin \mathcal{G}^-$ whp, and $\text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\} = \text{span}\{\mathcal{C}^- \cup \mathcal{D}^- \cup \{E\}\}$. Hence,

  $$\begin{aligned} b_S^+ &= \dim(\text{span}\{\mathcal{A}^- \cup \mathcal{B}^-\}) \\ &\quad - \dim(\text{span}\{\mathcal{A}^- \cup \mathcal{B}^-\} \cap \text{span}\{\mathcal{C}^- \cup \mathcal{D}^- \cup \{E\}\}) \\ &= b_S^- - 1 \end{aligned}$$

  Note that $\mathcal{G}^- \neq \text{span}\{\mathcal{A}^-\} \Leftrightarrow I_{\{i\}\to S^c}^- > 0$, and the existence of such $\lambda_l, \theta_k \Leftrightarrow I_{\{j\}\to S^c\setminus\{j\}}^- < m_j$.
  On the other hand, if $\mathcal{G}^- = \text{span}\{\mathcal{A}^-\}$, then $E \in \mathcal{G}^-$ and since $\mathcal{G}^+ = \mathcal{G}^-$, we will have $b_S^+ = b_S^-$.
- **Case 2**: Suppose no such λ_l, θ_k as in Case 1 exist. Let $\mathcal{F}^- = \{F_i^-, i \in [|\mathcal{F}^-|]\}$ be a basis for $\text{span}\{\mathcal{A}^- \cup \mathcal{B}^-\} \cap \text{span}\{\mathcal{C}^- \cup \mathcal{D}^-\}$ with $F_l^- = \sum_k \gamma_{lk} C_k^- + \sum_{k'} \mu_{lk'} D_{k'}^-$. Also, let $\mathcal{F}^+ = \{F_1^+, F_2^+, \ldots, F_{|\mathcal{F}^-|}^+\}$, where

  $$F_l^+ = F_l^- + (\sum_k \gamma_{lk}\beta_k) E, \; l \in \{1, 2, \ldots, |\mathcal{F}^-|\}. \quad (6)$$

  Note that $F_l^+ \in \text{span}\{\mathcal{A}^+ \cup \mathcal{B}^+\} \cap \text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\}$.
  Suppose $x \in \text{span}\{\mathcal{A}^+ \cup \mathcal{B}^+\} \cap \text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\}$, then there exists representations of x as follows.

  $$x = \sum_k \eta_k A_k^- + \sum_{k'} \delta_{k'} B_{k'}^- = \sum_l \xi_l (C_l^- + \beta_l E) + \sum_{l'} \zeta_{l'} D_{l'}^-$$

  Therefore, we have

  $$x - (\sum_l \xi_l \beta_l) E \in \text{span}\{\mathcal{A}^- \cup \mathcal{B}^-\} \cap \text{span}\{\mathcal{C}^- \cup \mathcal{D}^-\}$$

  $$\Rightarrow x - (\sum_l \xi_l \beta_l) E = \sum_l \tau_l F_l^- = \sum_l \tau_l (F_l^+ - \sum_k \gamma_{lk}\beta_k) E)$$

  Therefore,

  $$x - \sum_l \tau_l F_l^+ = \Big(\sum_l \xi_l \beta_l - \sum_{k,l} \tau_l \gamma_{lk} \beta_k\Big) E = \Phi(x) E \quad (7)$$

  We consider two cases here.
  Sub-case 2a: First, suppose that $\Phi(x) = 0$ for all $x \in \text{span}\{\mathcal{A}^+ \cup \mathcal{B}^+\} \cap \text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\}$. Hence, $\text{span}\{\mathcal{F}^+\} = \text{span}\{\mathcal{A}^+ \cup \mathcal{B}^+\} \cap \text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\}$. Next, we prove that members of $\mathcal{F}^+$ are linearly independent. Suppose $\sum_l \omega_l F_l^+ = 0$, then by (6),

  $$\sum_l \omega_l F_l^- = \Big(\sum_{l,k} \omega_l \lambda_{lk} \beta_k\Big) E \quad (8)$$

  Here, if $\mathcal{G}^- \neq \text{span}\{\mathcal{A}^-\}$, then $E \notin \mathcal{F}^-$ whp, and $\mathcal{F}^+$ are linearly independent, again whp. On the other hand, if $\mathcal{G}^- = \text{span}\{\mathcal{A}^-\}$, then $E \in \mathcal{F}^-$ can be uniquely represented as a linear combination of F_i^-, $i \in [|\mathcal{F}^-|]$. Let $E = \sum_l \psi_l F_l^-$. Given a particular value of $(\omega_1, \cdots, \omega_{|\mathcal{F}^-|}) \neq \mathbf{0}$, due to the randomness of the β_k's, the probability that $\sum_l \omega_l F_l^+ = 0$ happens is equal to $\frac{1}{q-1}$ which can be made as small as required by choosing a large field size.
  Thus, $\mathcal{F}^+$ are linearly independent in this case. Therefore,

  $$\dim(\text{span}\{\mathcal{A}^+ \cup \mathcal{B}^+\}) = \dim(\text{span}\{F^+\}) = \dim(\text{span}\{F^-\}).$$

  Therefore, the update rule will be $b_S^+ = b_S^-$.
  Sub-case 2b: suppose that $\Phi(x) \neq 0$ for some $x \in \text{span}\{\mathcal{A}^+ \cup \mathcal{B}^+\} \cap \text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\}$. Then, from (7), $E \in \text{span}\{\mathcal{A}^+ \cup \mathcal{B}^+\} \cap \text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\}$. Now, if $\mathcal{G}^- = \text{span}\{\mathcal{A}^-\}$, then $E \in \text{span}\{\mathcal{C}^- \cup \mathcal{D}^-\}$ which means that $\text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\} = \text{span}\{\mathcal{C}^- \cup \mathcal{D}^-\}$. Thus, the update rule in this case is given by $b_S^+ = b_S^-$. On the other hand, if $\mathcal{G}^- \neq \text{span}\{\mathcal{A}^-\}$, then $E \notin \text{span}\{\mathcal{C}^- \cup \mathcal{D}^-\}$. However, by (7), $E \in \text{span}\{\mathcal{C}^+ \cup \mathcal{D}^+\}$. Hence, there exists a representation of E as follows

  $$E = \sum_l \pi_l (C_l^- + \beta_l E) + \sum_{l'} \varphi_{l'} D_{l'}^- \quad (9)$$

  $$\Rightarrow \left(1 - \sum_l \pi_l \beta_l\right) E = \sum_l \pi_l C_l^- + \sum_{l'} \varphi_{l'} D_{l'}^-. \quad (10)$$

  Given that $E \notin \text{span}\{\mathcal{C}^- \cup \mathcal{D}^-\}$, it follows from (10) that $\sum_l \pi_l \beta_l = 1$ which implies that

  $$\sum_l \pi_l C_l^- + \sum_{l'} \varphi_{l'} D_{l'}^- = 0. \quad (11)$$

  However, in Case 2, there cannot be an equation of the form (11), unless we have $\pi_l = 0$ for all l. Substituting $\pi_l = 0$ in (9) results in a contradiction. Thus, Sub-case 2b occurs wlp. ■