Random Linear Network Coding for Time-Division Duplexing: Field Size Considerations

Random Linear Network Coding for Time-Division Duplexing: Field Size Considerations

Daniel E. Lucani
RLE, MIT
Cambridge, Massachusetts, 02139
Email: dlucani@mit.edu

Muriel Médard
RLE, MIT
Cambridge, Massachusetts, 02139
Email: medard@mit.edu

Milica Stojanovic
Northeastern University
Boston, Massachusetts, 02115
Email: millitsa@mit.edu

***Abstract*—We study the effect of the field size on the performance of random linear network coding for time division duplexing channels proposed in [1]. In particular, we study the case of a node broadcasting to several receivers. We show that the effect of the field size can be included in the transition probabilities of the Markov chain model of the system. Also, an improved upper bound on the mean number of coded packets required to decode M original data packets using random linear network coding is presented. This bound shows that even if the field size is 2, i.e. we perform XORs amongst randomly selected packets from the pool of M original ones, we will need on average at most $M+2$ coded packets in order to decode. Thus, there will be only a very small degradation in performance if M is large. We present numerical results showing that the mean completion time of our scheme with a field size of 2 is close in performance to our scheme when we use larger field sizes. We also show that as M increases, the difference between using a field size of 2 and larger field sizes decreases. Finally, we show that we can get very close to the optimal performance with small field sizes, e.g. a field size of 4 or 8, even when M is not very large.**

I. Introduction

Network coding was introduced by Ahlswede *et al* [2]. This concept is also known as coded packet networks. Network coding considers the nodes to have a set of functions that operate upon received or generated data packets. Today's networks constitute a subset of the coded packet networks, in which each node performs two main functions: forwarding and replicating a packet. A classical network's task is to transport packets provided by the source nodes unmodified. In contrast, network coding considers information as an algebraic entity, on which one can operate.

The use of network coding in time division duplexing (TDD) channels, i.e. when a node can transmit and receive, but not both at the same time [1]. The main insight provided by Reference [1] is that the transmitter should *vary* the amount of time allocated to transmit data and receive acknowledgements (ACK), based on the propagation time of the packets, the transmission time of the data and ACK packets, and the probability of erasures of the packets.

This scheme is rateless in nature. However, the feedback mechanism is different to that of typical rateless schemes, which assume an independent feedback channel through which the transmitter can receive an ACK indicating full decoding of the data. The scheme presented in [1] assumes that there is only one channel for both data and ACK, i.e. a TDD channel. Thus,

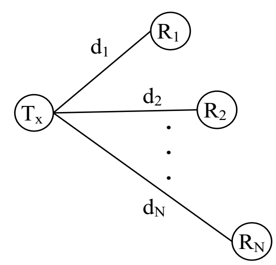

Fig. 1. [4] Broadcast network.

the transmitter will have to stop transmitting in order to listen for an ACK. Other rateless codes, e.g. LT codes [7], Raptor Codes [8], will face a similar challenge under the TDD channel, namely how many coded packets to send before stopping to listen for an ACK. The advantage of random linear network coding is that we can extend these ideas to general networks with no modifications from the coding perspective. Fountain codes, e.g. [7] [8], do not share this trade: intermediate nodes have to either relay packets with no modifications or completely decode the information before transmitting to other nodes in order to preserve the structure of the code.

In particular, Reference [1] studied the case of a node that has to transmit a block of M data packets through a link to another node using random linear network coding. This reference showed that there is an optimal number of coded data packets to be transmitted back-to-back before stopping to wait for an ACK, in terms of the mean time to complete transmission of the block of packets. Reference [3] extended this analysis to the problem of energy consumption of the scheme, showing that an optimal number of coded packets to be transmitted exists, under the minimum energy criterion. This reference also showed that choosing the number of coded data packets to optimize mean completion time, as in [1], provides a very good trade-off between energy consumption and completion time. Reference [4] provides an extension to the case of broadcast. In this setting, a transmitter with M data packets has the objective to broadcast those packets reliably to N receivers. This reference assumes that the receivers are not allowed to cooperate to share their received coded packets in order to decode, i.e. each receiver must decode the information from the coded packets sent directly from the transmitter.

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

These previous references have considered that the field size is large enough so that any random linear coded packet received was independent from previously received packets with very high probability. These references used field sizes of 1048576, which translates into using coefficients of 20 bits. We analyze the effect of the field size on our scheme, in the case of a node broadcasting to several receivers, as in Figure 1. The link case studied in [1] is a subset of the broadcast problem.

Also, we improve the upper bound on the mean number of coded packets required to decode M original data packets using random linear network coding presented in Reference [5]. We prove that even if the field size is 2, i.e. the transmitter is performing XORs amongst randomly selected packets from the pool of M original packets, each receiver will need on the average not more than $M+2$ coded packets in order to decode. Thus, there will be only a very small degradation in performance if M is large. This bound is valid for random linear network coding and independent on the TDD problem.

Numerical results compare the performance of our scheme when using the smallest possible field size $q = 2$ and when using much larger field sizes. We also illustrate that the difference in performance between a field size of 2 and larger field sizes decreases as M increases, as our bound suggested. Finally, if M is not very large, we show that 1) $q = 2$ has only a small degradation in performance, and 2) it is possible reduce considerably the gap to the optimal performance with very small field sizes, e.g. a field size of 4 or 8.

The paper is organized as follows. In Section II, we discuss the effect of the field size on the mean number of coded packets required to successfully decode the original M data packets. In Section III, we outline the set up of the problem and we study the mean completion time of our scheme considering the effect of the field size. Section IV provides numerical results for different link and broadcast scenarios. Conclusions are summarized in Section V.

II. Mean Number of Coded Packets Required for Successful Decoding

We can model the process of decoding M packets from the random linear coded packets received at a node as a Markov chain, as in Figure 2. A transition occurs when a new coded packet is successfully received at a node. This process repeats at every receiver.

The transition probability matrix for this problem is

$$P_q = \begin{bmatrix} q^{-M} & 1-q^{-M} & 0 & \cdots & 0 & 0 \\ 0 & q^{-M+1} & 1-q^{-M+1} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ 0 & 0 & 0 & \cdots & q^{-1} & 1-q^{-1} \\ 0 & 0 & 0 & \cdot\cdot & 0 & 1 \end{bmatrix}.$$

Let us provide a full characterization of this problem by obtaining the moment generating function of the number of coded packets that need to be received before successfully decoding the information. We state this result in the following lemma.

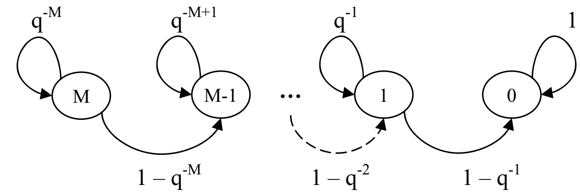

Fig. 2. Markov Chain of the degrees of freedom required to decode. Transitions occur when a new coded packet is successfully received by a node.

Lemma 1: The moment generating function $M_n(s)$ of the number of coded packets that need to be received before successfully decoding all the data when n linearly independent coded packets are needed to decode is given by

$$M_n(s) = \frac{e^s}{1 - P_{n\to n}e^s} P_{n\to(n-1)} M_{n-1}(s) \tag{1}$$

with $M_0(s) = 1$

Proof: It follows the same steps as the proof in [9]. ■

Let us bound the average number of coded packets that need to be received before successfully decoding the M packets. Clearly, at least M coded packets must be received before being able to decode. Thus, a trivial lower bound is M. The upper bound is given by the following lemma.

Lemma 2: If M data packets are encoded using random linear network coding with a field size q, then the mean number of coded packets that have to be received before completely decoding the original packets is upper bounded by

$$\min\left\{ M\frac{q}{q-1}, M+1+\frac{1-q^{-M+1}}{q-1} \right\} \tag{2}$$

Proof: Let us define the minimum number of coded packets received to decode as N_c. Then $E[N_c] = \sum_{k=1}^{M} \frac{1}{1-q^{-k}}$.

Since $q^{-k} \leq q$ for $q \geq 2$ and $k \geq 1$, then $E[N_c] \leq \sum_{k=1}^{M} \frac{q}{q-1} = M\frac{q}{q-1}$ which shows the first bound, proved in Reference [5].

The second bound comes from

$$E[N_c] = M + \sum_{k=1}^{M} \frac{1}{q^k - 1} \leq M + \sum_{k=0}^{M-1} q^{-k} \tag{3}$$

$$= M + \frac{1-q^{-M}}{1-q^{-1}} = M + 1 + \frac{1-q^{-M+1}}{q-1} \tag{4}$$

where we have used the fact that $q^k - 1 \geq q^{k-1}$ for $k \geq 1$ and $q \geq 2$. ■

One important conclusion of this lemma is that $E[N_c] \leq M + 2, \forall q \geq 2$, i.e. on average the number of coded packets needed to decode the M original packets will be between M and $M+2$ for any field size. Note that if $M >> 2$ we expect that a scheme using $q = 2$ and one using larger q will have a small difference in performance.

Figure 3 illustrates the upper and lower bounds for a wide range of field sizes. It also shows that the upper bound $E[N_c] \leq M\frac{q}{q-1}$ becomes the dominant bound for large q, while the $E[N_c] \leq M+1+\frac{1-q^{-M+1}}{q-1}$ is the dominant bound for small

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

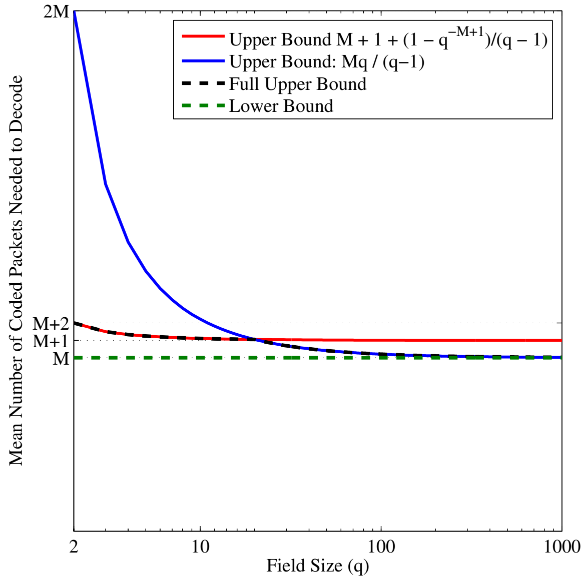

Fig. 3. Upper and Lower bounds on the mean number of coded packets needed at the receiver in order to decode versus the field size q.

q. Finally, Lemma 3 shows that the new bound on $E[N_c]$, i.e. $E[N_c] \leq M + 1 + \frac{1-q^{-M+1}}{q-1}$, will be the dominant bound up to a field size q that increases as M increases.

Lemma 3: Exists q^* for which $M\frac{q}{q-1} \leq M + 1 + \frac{1-q^{-M+1}}{q-1}, \forall q \geq q^*, M \geq 1$. q^* scales as $O(M)$.

Proof: Since both bounds are decreasing functions and $\lim_{q\to\infty} M\frac{q}{q-1} = M$ and $\lim_{q\to\infty} M + 1 + \frac{1-q^{-M+1}}{q-1} = M+1$, q^* exists. Note that $M+1+\frac{1-q^{-M+1}}{q-1} \in [M+1, M+2], \forall q \geq 2$. Thus, the bounds must cross in this region. Let us consider $\alpha \in [1,2]$ and find q_α that satisfies $M\frac{q_\alpha}{q_\alpha - 1} = M+\alpha$, i.e. $q_\alpha = M/\alpha + 1$. Thus, $q_2 = M/2+1 \leq q^* \leq M+1 = q_1$, which concludes the proof. ■

III. Random Network Coding for Broadcast in TDD Channels

We study the case of one node broadcasting information to N receivers. We consider independent erasure channels for each receiver, as in [4], and follow a similar analysis. Our main contribution is to include the effect of the field size into the transition probabilities. These transition probabilities are considerably different from those used in [4], [1] and [3]. We present an overview of the scheme before starting our study of the transition probabilities.

We consider that a sender wants to broadcast M data packets at a given data rate R [bps] to N receivers as in Figure 1. We assume that receivers cannot cooperate or share information, which means that they can only obtain their information from the sender. Since we have a TDD constraint, nodes can transmit and receive, but not both at the same time. We assume that the sender uses random linear network coding [6] to generate coded data packets. Each coded data packet has three parts: 1) an information header of size h bits, 2) data section which

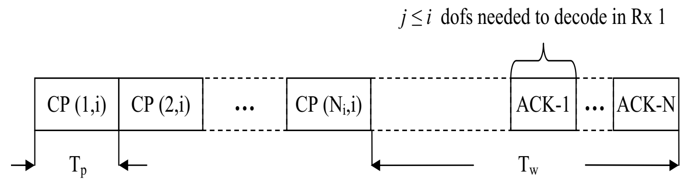

Fig. 4. [4] Network coding TDD scheme for broadcast.

contains a random linear combination of the M data packets of n bits each, and 3) the random coding coefficients used in the linear combination of the data packets. If we are encoding over a field size q, then each coefficient is represented by g bits, where $g = \log_2 q$. Thus, the total number of bits per packet is $h + n + gM$ [4]. Figure 2 in [1] shows the structure of each coded packet.

Similarly as in[4], the sender can transmit coded packets back-to-back before stopping to wait for an ACK packet from each receiver. Every ACK packet returns the number of degrees of freedom (dof) that a particular receiver still requires to decode successfully the M original data packets.

The transmission process starts with M data packets being encoded into $N_M \geq M$ random linear coded packets, and broadcasted to the N receivers. If all M packets are decoded successfully by all receivers, the process is completed. Otherwise, each receiver sends an ACK packet that informs the transmitter how many dofs are missing, say $i_1, i_2, ..., i_N$ for receivers $1, 2, ..., N$, respectively. At this point, the transmitter sends N_i coded packets, where $i = \max_{j=1,2,...,N} i_j$, as in [4]. The process is repeated until the M data packets are successfully decoded by all receivers. As in previous work, we are interested in the optimal number N_i of coded packets to be transmitted back-to-back in order to minimize a specific metric, e.g. mean completion time.

Figure 4, illustrates the time window allocated to the system to transmit N_i coded packets. Note that each coded packet $CP(1,i)$, $CP(2,i)$, etc. takes T_p time units to be transmitted. The waiting time T_w is chosen so as to accommodate the propagation delay and time to receive the ACKs from each receiver [4].

We showed in [4] that this process can be modelled as a Markov chain, where each state $(s_1, s_2, ..., s_N)$ is defined by the number of dofs required, s_k at receiver k, to decode successfully the M packets. The states range from $(M, M, ..., M)$ to $(0, 0, ..., 0)$. This is a Markov chain with $(M+1)^N - 1$ transient states and one recurrent state (state $(0, 0, ..., 0)$) [4].

Let us compute the transition probabilities considering the effect of the field size. The transition probabilities from state $(s_1, s_2, ..., s_N)$ to state $(s'_1, s'_2, ..., s'_N)$ are

$$P_{(s_1,s_2,...,s_N)\to(s'_1,s'_2,...,s'_N)} =$$
$$P\left(X_1(n){=}s'_1,...,X_N(n){=}s'_N | X_1(n-1){=}s_1,...,X_N(n-1){=}s_N\right)$$

where $X_i(n)$ is the number of dof required at receiver i at the end of transmission n. For simplicity of notation, let us say that $P\left(X_1(n){=}s'_1,...,X_N(n){=}s'_N | X_1(n-1){=}s_1,...,X_N(n-1){=}s_N\right) =$

$P\left(s'_1,...,s'_N|s_1,...,s_N\right)$. Similarly, we consider that $P\left(X_i(n)=s'_i|X_1(n-1)=s_1,...,X_N(n-1)=s_N\right) = P\left(s'_i|s_1,...,s_N\right)$ and $P\left(X_i(n)=s'_i|X_i(n-1)=s_i,\max_{j=1,2,...,N} s_j\right) = P\left(s'_i|s_i,\max_{j=1,2,...,N} s_j\right)$.

If we consider independent packet erasure channels for each of the receivers,

$$P_{(s_1,...,s_N)\to(s'_1,...,s'_N)} = P\left(s'_1|s_1,...,s_N\right)...P\left(s'_N|s_1,...,s_N\right).$$

The dependence on the previous state $(s_1, s_2, ..., s_N)$ can be translated into a dependence on the state with maximum dofs required to transmit, i.e. $i = \max_{j=1,2,...,N} s_j$, because i determines N_i, the number of coded data packets sent by the transmitter. Thus,

$$P_{(s_1,s_2,...,s_N)\to(s'_1,s'_2,...,s'_N)} =$$
$$P\left(s'_1|s_1,\max_{j=1,2,...,N} s_j\right)...P\left(s'_N|s_N,\max_{j=1,2,...,N} s_j\right) =$$
$$P\left(s'_1|s_1,N_i\right)...P\left(s'_N|s_N,N_i\right).$$

Let us study in detail the probabilities $P\left(s'_j|s_j,N_i\right)$. We assume that $N_i \geq s_j$, which means that there is some probability of transitioning from any s_j to $s'_j = 0$. For $s_j > s'_j$ we have that

$$P\left[s'_j|s_j,N_i\right] =$$
$$= (1 - Pe_{ack-j}) \sum_{k=\max\{1,s_j-s'_j\}}^{N_i} P\left[s'_j|s_j,k\right] P\left[k|N_i\right]$$

where we have used the fact that regardless of what combinations the transmitter sent N_i, the transition at each receiver will only depend on the number of coded packets that have been received k, i.e. $P\left[s'_j|s_j,N_i,k\right] = P\left[s'_j|s_j,k\right]$.

Note that $P\left[s'_j|s_j,k\right]$ represents the probability of starting at state s_j and transitioning to state s'_j in k transitions or hops. But this can be found by computing P_q^k, using the P_q computed in the previous section, and searching in the appropriate column and row corresponding to starting state s_j and end state s'_j.

For the case of $s_j = s'_j > 0$,

$$P\left(s_j|s_1,N_i\right) =$$
$$(1 - Pe_{ack-j}) \left[\sum_{k=0}^{N_i} P\left[s_j|s_j,k\right] P\left[k|N_i\right]\right] + Pe_{ack-j}$$

and that $P\left(0|0,N_i\right) = 1$. Finally, note that $P\left[k|N_i\right] = \binom{N_i}{k}(1 - Pe_j)^k Pe_j^{N_i-k}$ which completes the characterization of the problem.

A. Mean Completion Time

The mean time for completing the transmission of the M data packets to all receivers constitutes the expected time of absorption, i.e. the time to reach state $(0, ..., 0)$ for the first time, given that the initial state is $(M, ..., M)$. Reference [4] studies this problem in more detail.

We can define T^i as the time it takes to transmit N_i coded data packets and receive the ACK packets from the different receivers. It is easy to show that $T^i = N_i T_p + T_w$, where T_w is described in [4].

The mean completion time when the system is in state $(s_1, ..., s_N)$ is given by

$$T_{(s_1,...,s_N)} = T^i +$$
$$\sum_{(s_1,...,s_N),(s'_1,...,s'_N)} P_{(s_1,...,s_N)\to(s'_1,...,s'_N)} T_{(s'_1,...,s'_N)}$$

where $i = \max_{j=1,...,N} s_j$. We can express this in vector form as

$$\bar{T} = [I - P]^{-1}\bar{\mu}. \tag{5}$$

where $\bar{T} = [T_{(s_1,...,s_N)}]$, $\bar{\mu} = [T^i]$ and P is the corresponding transition probability.

Since we are interested in the mean completion time when we start at state $(M, ..., M)$, we can use Cramer's rule as in Reference [4] to determine

$$T_{(M,...,M)} = \frac{\det\left(\Gamma \leftarrow_{(M,...,M)} \bar{\mu}\right)}{\det(\Gamma)} \tag{6}$$

where $\Gamma = I - P$, and the notation $\Gamma \leftarrow_{(M,...,M)} \bar{\mu}$ represents a matrix that has all columns as the Γ matrix except the column corresponding to state $(M, ..., M)$ which is substituted by the vector $\bar{\mu}$. Due to characteristics of the Markov chain, Γ is a triangular matrix. Thus, computing $\det(\Gamma)$ reduces to multiplying the elements in the main diagonal of the Γ matrix.

B. Minimizing Mean Completion Time: Single Receiver

Our objective is to minimize the value of the expected transmission time T_M.

$$\min_{N_M,..,N_1} T_M = \min_{N_M} \frac{T^M + \sum_{i=1}^{M-1} P_{M\to i} \min_{N_i,..,N_1} T_i}{1 - P_{M\to M}} \tag{7}$$

where $T^i = N_i T_p + T_w$. Similar to the result in [1], regardless of the assumption on N_i, the problem of minimizing T_M in terms of the variables $N_M, .., N_1$ can be solved iteratively. First, we compute $\min_{N_1} T_1$, then use this results in the computation of $\min_{N_2,N_1} T_2$, and so on. Thus, we can preserve the search method proposed in [1] to find the optimal value of the N_i's. This search method exploited the recursive characteristic of the problem, to transform a M-dimensional integer search to M one-dimensional integer searches.

C. Minimizing Mean Completion Time: Multiple Receivers

The problem of optimizing the N_i's for the multiple receiver case is more complicated than the single receiver case. As explained in [4], there are $(M + 1)^N$ states in our Markov chain. This means that for each iteration of a full search algorithm we would have to compute the transition probabilities to fill a $\left((M + 1)^N - 1\right)$ x $\left((M + 1)^N - 1\right)$ matrix, and then solve the determinants of matrices of the same dimensions. Thus, the computational demands increase significantly, specially as the number of receivers increases.

For this reason, Reference [4] considered some heuristics to estimate the values of $N_i, \forall i = 1, ..., M$. These heuristics relied on solving the link case considering as packet erasure probability of the link a function of the packet erasure probabilities of the different channels in broadcast. The best heuristic

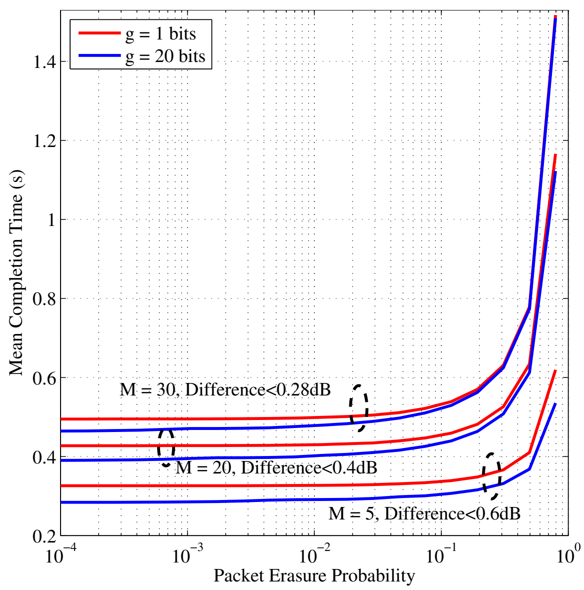

Fig. 5. Mean completion time for the TDD scheme with different M and field sizes $q=2$ and $q=1048576$, for $g=1$ and $g=20$, respectively. We use the following parameters $R=1.5$ Mbps, $h=80$ bits, $n_{ack}=100$ bits.

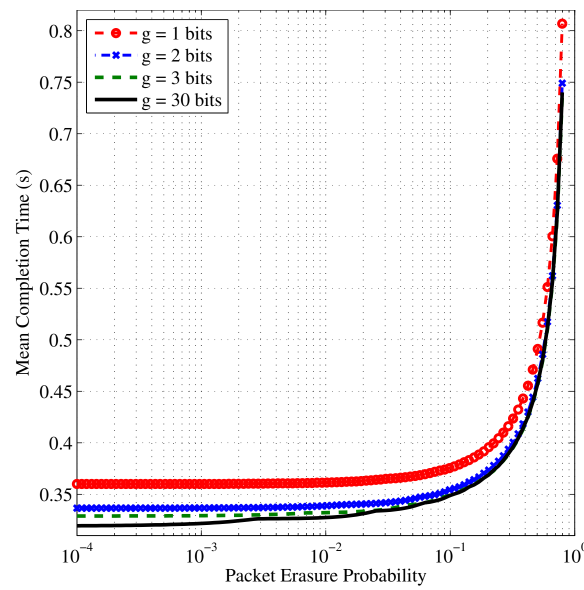

Fig. 6. Mean completion time for the TDD scheme with a single receiver with different field sizes $q=2^g$. We use the following parameters $R=1.5$ Mbps, $h=80$ bits, $n_{ack}=100$ bits, $M=10$.

was called the 'Worst Link Channel' heuristic. In this heuristic we approximated the system as a link to the receiver with the worst channel, i.e. $Pe=\max_j Pe_j$. Then, we computed $N_i, \forall i = 1,...,M$ to minimize the mean completion time using the values of T_p, T_w for the broadcast problem, and $Pe_{ack}=\max_j Pe_{ack-j}$. Note that for the choice of the N_i's we must use the transition probabilities studied in this paper, in order to consider the effect of the field size.

IV. Numerical Results

This section provides numerical results that compare the performance of our network coding scheme in TDD channels, considering the effect of different field size. We consider a GEO satellite setting with a propagation time $T_{prop}=125$ ms [1], and data packets of size $n=10,000$ bits. We compare performance of the scheme in terms of mean completion time under different packet erasure probabilities. We show that using $q=2$ shows a small degradation in performance with respect to higher field sizes. Also, the gap in performance between $q=2$ and higher q reduces as M increases, as expected. Finally, if the performance of $q=2$ is not sufficiently good for small M, we can get very close to the performance of high field sizes with small increases of the field size, e.g. $q=4$ or $q=8$.

Figure 5 shows the mean completion time for the TDD scheme for a single receiver for $q=2$ and $q=2^{20}$ for various block sizes M and a wide range of packet erasure probabilities. Figure 5 illustrates that the gap between field sizes $q=2$ and $q=2^{20}$ is very small. For $M=5$ the gap is smaller than 0.6 dB for the range of packet erasure probabilities considered, which ranges from 10^{-4} to 0.8. This means that the completion time is increased by at most 15 % on average for $M=5$. For $M=20$ and $M=30$ we observe that the gap reduces to less than 0.4 dB and 0.28 dB, respectively. In other words, the

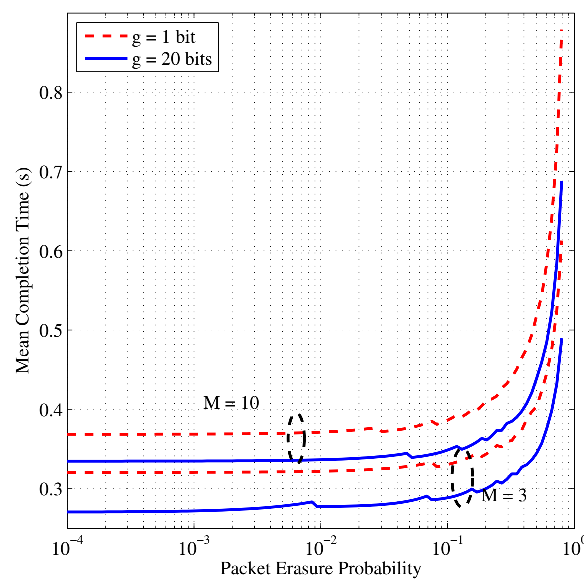

Fig. 7. Mean completion time for the TDD scheme with two receivers with different M and field sizes $q=2$ and $q=1048576$, for $g=1$ and $g=20$ bits, respectively. We use the following parameters $R=1.5$ Mbps, $h=80$ bits, $n_{ack}=100$ bits.

completion time is increased on average by at most 10 % and 6.6 %, respectively. The importance of this result is two-fold. First, the degradation in performance due to the use of $q=2$ is very small, even for small values of M where the effect of small field size is more noticeable. Also, the degradation in performance reduces as M, the number of data packets that are being randomly combined, increases. This effect was predicted by the result in Lemma 2. Since we expect to need between M and $M+2$ coded packets on average in order to decode, then the effect of the additional coded packets is clearly reduced if M increases because proportionally more resources are being used to transmit the first M coded packets than the additional coded packets needed to finally decode.

Second, we can rely on considerably simpler coders and decoders. Note that for $q = 2$, random linear network coding is basically performing an XOR of those packets that were chosen from the pool of M original packets. Note that each packet has a probability of $1/2$ to be chosen to be XORed in each coded packet that is being generated. Also, the overhead on the coded packet is reduced because the coefficient size $g = 1$ bit. For large enough M and n fixed, $q = 2$ could outperform cases where q is larger than 2 because larger q and larger M involve an increased overhead in the coded packet, i.e. cases in which we are using more resources sending information about coefficients than sending the actual information.

Figure 6 illustrates that if M is small, e.g. $M = 10$ in the figure, and the performance of $q = 2$ is insufficient, we can get considerable improvements with small field sizes. Figure 6 considers a single receiver and the cases of $q = 4$ and $q = 8$, which correspond to $g = 2$ and $g = 3$ bits, and compares it to the performance of $q = 2^{30}$, i.e. coefficients of $g = 30$ bits. Note that $q = 8$ is extremely close to the performance of $q = 2^{30}$, especially for $Pe > 0.01$ which is a range of common Pe values for wireless systems. We observe that for $Pe > 0.1$, the performance of $q = 4$ is essentially the same to that of our scheme using a field size of $q = 2^{30}$. Note that for a GEO satellite example the range of $Pe > 0.1$ are typical values. Thus, for wireless systems we could expect similar performance if we use small or large field sizes, even if M is not too large.

Figure 7 illustrates the case of a system with two receivers at the same distance from the transmitter, which is a good approximation in some satellite applications. We consider different small values of M and compute the N_i's using the 'Worst Link Channel' heuristic. We consider that the each receiver has an independent channel but that the packet erasure probability for each channel is the same, i.e. $Pe_1 = Pe_2 = Pe$, and that there are no erasures for the ACK packets. We observe that the gap between using a field size of $q = 2$ and $q = 2^{20}$ is again small, even for very small values of M. Note that for $M = 3$ the gap is always smaller than 0.74 dB or, equivalently, an increase of 18 % in the completion time when we use $q = 2$ with respect to $q = 2^{20}$. For $M = 10$ the gap is of 0.42 dB or, close to a 10 % increase when we use $q = 2$. Thus, we observe that the gap between $q = 2$ and $q = 2^{20}$ decreases as M increases.

V. Conclusion

This paper considers the effect of field size in random linear network coding over time division duplexing channels. We show that we can maintain the Markov chain models proposed in previous work, e.g. [1], [4], including the effect of the field size in the transition probabilities. We also showed that the search algorithm proposed in Reference [1] for a link is still valid when we consider the effect of the field size.

We provided bounds on $E[N_c]$,the mean number of coded packets that a receiver needs to receive successfully in order to decode the information. This bounds are valid for random linear network coding in general, i.e. without any assumption on the channel characteristics or the TDD constraint. A trivial lower bound is that $E[N_c] \geq M$. We prove an insightful upper bound, which states that $E[N_c] \leq \min\left\{M\frac{q}{q-1}, M + 1 + \frac{1-q^{-M+1}}{q-1}\right\}$. This bound implies that $E[N_c]$ can become arbitrarily close to M as q increases, but more importantly, we showed that $E[N_c] \leq M + 2$ for any $q \geq 2$. This means that as M increases, the effect of the additional coded packets that have to be sent due to a "bad" random selection of the coefficients, will be negligible. A "bad" random selection is a choice of coefficients that does not provide innovative information.

We present numerical results that illustrate that the gap between using $q = 2$ and larger values of q is small, specially when M is large. Note that we can rely on considerably simpler coders and decoders. For $q = 2$, random linear network coding is basically performing an XOR of those packets that were chosen from the pool of M original packets, each packet is chosen to be combined with probability $1/2$. Finally, if the performance of $q = 2$ is not sufficiently good for small M, we can get very close to the performance of high field sizes with small increases of the field size, e.g. $q = 4$ or $q = 8$.

Acknowledgment

This work was supported in part by the National Science Foundation under grants No. 0831728 and CNS-0627021, by ONR MURI Grant No. N00014-07-1-0738, subcontract # 060786 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. N66001-06-C-2020 (CBMANET), subcontract # 18870740-37362-C issued by Stanford University and supported by the DARPA.

References

[1] D. E. Lucani, M. Stojanovic, M. Médard, "Random Linear Network Coding For Time Division Duplexing: When To Stop Talking And Start Listening", in Proc. INFOCOM'09, Rio de Janeiro, Brazil, pp. 1800-1808, Apr. 2009

[2] R. Ahlswede, N. Cai, S. Y. R. Li, R. W. Yeung, "Network Information Flow", IEEE Trans. Inf. Theory, vol. 46, no. 4, pp. 1204-1216, Jul. 2000

[3] D. E. Lucani, M. Stojanovic, M. Médard, "Random Linear Network Coding For Time Division Duplexing: Energy Analysis", in Proc. ICC'09, Dresden, Germany, Jun. 2009

[4] D. E. Lucani, M. Médard, M. Stojanovic, "Broadcasting in Time-Division Duplexing: A Random Linear Network Coding Approach", in Proc. NetCod'09, Lausanne, Switzerland, pp. 62-67, Jun. 2009

[5] A. Eryilmaz, A. Ozdaglar, M. Médard, "On Delay Performance Gains from Network Coding", In Proc. CISS'06, Princeton, NJ, USA, pp. 864-870, Mar. 2006

[6] T. Ho, M. Medard, R. Koetter, D. R. Karger, M. Effros, J. Shi, B. Leong,"A Random Linear Network Coding Approach to Multicast", Trans. Info. Theory, vol. 52, no. 10, pp.4413-4430, Oct. 2006

[7] M. Luby,"LT-codes," In Proc. 43rd Annu. IEEE FOCS, Vancouver, BC, Canada, pp. 271280, Nov. 2002.

[8] A. Shokrollahi, "Raptor Codes," IEEE Trans. Info. Theory, vol. 52, no. 6, pp. 25512567, 2006

[9] D. E. Lucani, M. Médard, M. Stojanovic, "On coding for delay - New approaches based on network coding in networks with large latency," in Proc. ITA Workshop, San Diego, CA, pp. 191-200, Feb. 2009