Asymptotic Rate Limits for Randomized Broadcasting with Network Coding

Di Niu, Baochun Li

Department of Electrical and Computer Engineering
University of Toronto
$\{$*dniu, bli*$\}$*@eecg.toronto.edu*

***Abstract*—Motivated by peer-to-peer content distribution and media streaming applications, we study the broadcasting problem in a time-discretized model, with integer valued upload and download capacity constraints at nodes. We analyze both deterministic centralized and randomized decentralized protocols that can achieve optimal packet receiving rates at the nodes. In particular, we consider a simple scheme that requires each node, in each time slot, to transmit to a random neighbor that is not yet chosen by any other nodes in that slot. We prove that such a surprisingly simple scheme can asymptotically achieve the optimal receiving rates in complete graphs with homogeneous node capacity. The proof involves applying randomized network coding and deriving the max-flow bounds achieved in the resulting transmission schedule. We extend the results to more general topologies, and bound the performance of randomized neighbor selection with randomized network coding.**

I. Introduction

Peer-to-peer content distribution and media streaming systems have gained enormous popularity in reality. For example, in peer-to-peer file sharing systems, a seed node may share bulk files to tens of thousands of other hosts. Peer-to-peer media streaming systems rely on clients' ability to store and forward media data generated at the server, so that the media can be played back at all the users at a required quality. Underlying the core of such applications is broadcasting, preferably performed in a decentralized and randomized fashion to achieve scalability and robustness.

Motivated by these applications, we consider the following node-capacitated broadcasting problem. Packets are being streamed from a single source node to all the other nodes in a connected undirected graph. Communication happens in synchronized time slots. Each node has both an upload capacity and download capacity, which limit the number of packets it can send or receive in each time slot. Since a packet is the minimum data unit in our problem, we assume both upload and download capacity take integer values. Each node has the freedom to decide which packet to upload to which peer in each time slot. Such transmission schedules may be generated at a centralized coordinating unit or at the nodes in a decentralized way.

We ask the question — under the above model, what is the maximum rate at which each node in the network can receive information? And what is the protocol that supports such rates? We first give two deterministic protocols that can achieve the optimal receiving rates at nodes. However, these protocols require a high degree of centralized coordination, and thus are not easy to be implemented. They are also too rigid to adapt to network topology and condition changes, and incur unfair delay performance at users, i.e., some users always receive packets later than other users.

To tackle these problems, we further ask the question — is there any sufficiently simple decentralized protocol that requires the least possible control overhead and yet still achieves the optimal rates? We give an affirmative answer to this question by considering a very simple protocol: in each time slot, each node uploads to a random neighbor that has not yet been chosen by any other node in that time slot. We prove the surprising result that in complete graphs with homogeneous node capacity, this simple random neighbor selection scheme can asymptotically achieve the optimal rates at all nodes in the long run. The proof involves applying randomized network coding [1], and deriving the max-flow bound achieved by network coding [2], [3] on the resulted transmission trellis graph, using renewal reward theory [4].

As a by-product, we have shown that simply by letting each node forward the packet received in the previous time slot (latest packet first), approximately one half of the optimal rate can be achieved at each node, with the random neighbor selection scheme. We also extend to analyze a more general class of graphs which have at least one Hamiltonian cycle, and derive the exact rate achieved by latest packet first. This forms a lower bound on the optimal rates that random neighbor selection combined with randomized network coding can asymptotically achieve.

The remainder of the paper is organized as follows. Sec. II reviews the related work. The problem is formulated in Sec. III. Sec. IV presents two rate-optimal deterministic protocols, followed by an analysis of their delay-performance. We analyze the simple decentralized protocol in Sec. V and prove its rate-optimality. In Sec. VI, we extend our analysis to networks with Hamiltonian cycles, and demonstrate the usefulness of our results with examples. Sec. VII concludes the paper and discusses the future work.

II. Related Work

Broadcasting in a directed network with edge capacity constraints is a well studied problem. Edmonds [5] shows that the optimal broadcast rate for a directed graph can be achieved by packing edge-disjoint spanning trees rooted at the source. However, such an algorithm is centralized in nature and is unsuitable for peer-to-peer content dissemination. Closely related to our work is [6], presented by Massoulie *et al.*, who also study node-capacitated broadcasting problems with decentralized algorithms. They have shown that random useful packet forwarding combined with most deprived neighbor selection achieves the optimal broadcast rate in complete networks, if each packet transmission takes exponential time.

Our work is different from [6] in the following ways. The neighbor selection and packet selection in [6] heavily depend on each other, and thus incur a high degree of message passing and coordination in the network. In contrast, we study neighbor selection and packet selection (coding) algorithms that are *completely decoupled*. In particular, the much simpler random neighbor selection only requires the minimum control overhead. Furthermore, adopting a discrete-time model, our results are not contingent on the assumption of exponential packet transmission time. In addition, our optimality result is proved under both upload and download capacity constraints, while in [6] the download capacity of each node is assumed to be infinite.

The delay of broadcasting k packets using decentralized protocols in a discrete time model has also been studied. Yeung [7] shows that network coding achieves the optimal delay to broadcast k packets for any neighbor selection schedules. Deb *et al.* [8] show that randomized network coding can achieve a shorter broadcast delay of k packets in complete graphs, as compared to a naive sequential dissemination. Mosk-Aoyama *et al.* [9] further analyze the broadcast delay using network coding in arbitrary graphs and show its correlation with the spectral properties of the graph. Sanghavi *et al.* [10] consider the problem of broadcasting multiple packets in P2P networks, and propose a decentralized packet exchange algorithm based on pushes and pulls that has a close-to-optimal performance.

III. Problem Formulation

Consider the network as an undirected graph $G = (V, E)$. There is one source node s, and $N = |V| - 1$ sinks that wish to receive the same data from s. We assume the broadcast happens in synchronized time slots in the following manner:

- Each node can only transmit to its neighbors.
- Each node v has an integer upload capacity of U_v, and an integer download capacity of D_v. Specifically, in each time slot, node v can upload U packets with $U \in \{0, 1, \ldots, U_v\}$, and can download D packets with $D \in \{0, 1, \ldots, D_v\}$.

Each node can only upload an integer number of packets at each time, because a packet is the minimum data unit in our problem. In this paper, we only consider the case of homogeneous node capacity. Without loss of generality, we assume $U_v = D_v = 1$, $\forall v \in V$.

A **routing protocol** decides for each node what packets to transmit to which receivers at time t. Alternatively, a routing protocol can be viewed as a combination of a **receiver selection protocol** and a **network code** that might or might not be inter-dependent. The receiver selection protocol decides the sender-receiver pairs in each time slot subject to the capacity constraints, while the network code determines for each sender what packet to transmit (with the encoding of packets allowed) in each time slot.

Let $R_v(t)$ be the number of new packets that node v receives in slot t. Define the time average receiving rate of node v as

$$\mathcal{R}_v = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=1}^{t} R_v(\tau) \qquad (1)$$

Let $T_v(p)$ denote the delay from the issuance of packet p at source s to the reception of p at node v, e.g., if s transmits a new packet p to v_1 at time t, and v_1 transmits p to v_2 at time $t+1$, then $T_{v_1}(p) = 1$, $T_{v_2}(p) = 2$. Let $\mathcal{P}_v(t)$ be the set of packets received by node v by time t. We define the time average propagation delay at node v as

$$\mathcal{T}_v = \lim_{t\to\infty} \mathcal{T}_v(t) = \lim_{t\to\infty} \frac{1}{|\mathcal{P}_v(t)|} \sum_{p\in\mathcal{P}_v(t)} T_v(p) \qquad (2)$$

The goal of the broadcast problem is to find a routing schedule such that each node receives information at the maximum rate while maintaining the fairness with respect to the propagation delays at all nodes. For example, when the network is a complete graph, and $U_v = D_v = 1$ for all $v \in V$, one rate-optimal protocol is to let all the nodes form a chain, with s being the head and each node i ($i = 1, 2, \ldots, N-1$) forwarding the packet it has just received to node $i+1$. Clearly, this protocol does not perform well in terms of delay fairness: node i always suffers a delay of $\mathcal{T}_i = i$ hop(s).

Implementability is another important factor to consider when choosing protocols. Ideally, the receiver selection and the network code should be decoupled so that minimum control overhead is required in a decentralized setting.

IV. Optimal Deterministic Schedules

We first consider the case of complete graphs. In this section, we propose two deterministic routing protocols that can yield the optimal $\mathcal{R}_v$ for each node v while attempting to minimize $\max_v \mathcal{T}_v$.

Let $v_1, \ldots, v_N$ be a permutation of $1, \ldots, N$. We say a packet p takes the route $s \to v_1 \to \ldots \to v_N$, if s transmits p to v_1 at $t = 1$, v_1 transmits p to v_2 at $t = 2$, and so on. Since $U_v = D_v = 1$ for all $v \in V$, finding a rate-optimal schedule is equivalent to determining the routes for all the packets issued by s, subject to the capacity constraint.

Longest Delay First (LDF). The source s issues a new packet p_t at time t ($t = 1, 2, \ldots$). p_1 takes the route $s \to 1 \to \ldots \to N$. Given the routes of $p_1, \ldots, p_{t-1}$, packet p_t starts from s and chooses the jth ($j = 1, \ldots, N$) node v_j in its route one by one. v_j is a node such that the capacity

t	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
1	s									
2	1	s								
3	2	4	s							
4	3	2	1	s						
5	4	3	2	1	s					
6		1	3	4	2	s				
7			4	3	1	2	s			
8				2	3	4	1	s		
9					4	3	2	1	s	
10						1	3	4	2	s
11							4	3	1	2
12								2	3	4
13									4	3
										1

Fig. 1. A schedule given by LDF for $N = 4$. When $N = 4$, FBA generates the same schedule. However, FBA and LDF are different in general when N is even.

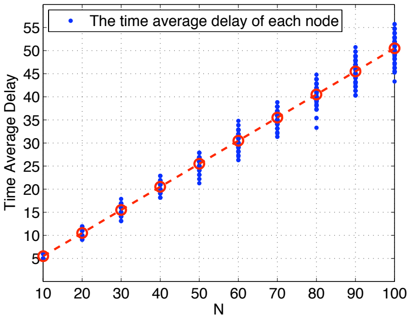

Fig. 2. The time average propagation delay of LDF for even N. Each small dot denotes the time average delay $\mathcal{T}_v$ of each node v for the corresponding N. The circle denotes the average across all nodes: $\frac{1}{N}\sum_{v\in V\backslash s}\mathcal{T}_v = (N+1)/2$. For each N, we run the simulation for $20N$ rounds to get the time average.

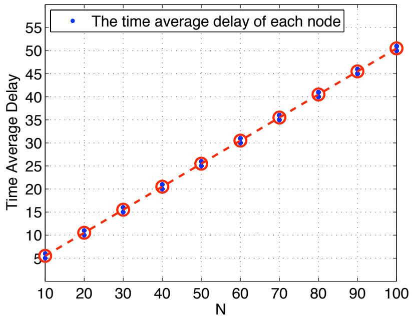

Fig. 3. The time average propagation delay of FBA for even N. Each small dot denotes the time average delay $\mathcal{T}_v$ of each node v for the corresponding N. The circle denotes the average across all nodes: $\frac{1}{N}\sum_{v\in V\backslash s}\mathcal{T}_v = (N+1)/2$.

constraint $U_{v_j} = D_{v_j} = 1$ is not violated at time $t+j$, and

$$v_j = \arg\max_{v:v\notin\{s,v_1,\ldots,v_{j-1}\}} \sum_{\tau=1}^{t-1} T_v(p_\tau).$$

Forward Backward Alternate (FBA). The source s issues a new packet p_t at time t ($t = 1, 2, \ldots$). p_1 takes the route $s \to 1 \to \ldots \to N$. Given the routes of $p_1, \ldots, p_{t-1}$, packet p_t starts from s and chooses the jth ($j = 1, \ldots, N$) node v_j in its route one by one. v_j is a node such that the capacity constraint $U_{v_j} = D_{v_j} = 1$ is not violated at time $t+j$, and

$$v_j = \begin{cases} \arg\max_{i:1\le i\le N, i\notin\{v_1,\ldots,v_{j-1}\}} i, & \text{for } p_t \text{ with even } t,\\ \arg\min_{i:1\le i\le N, i\notin\{v_1,\ldots,v_{j-1}\}} i, & \text{for } p_t \text{ with odd } t.\end{cases}$$

It is easy to check that when N is odd, LDF and FBA will generate the same schedule, i.e., packet p_t will take the route $s \to 1 \to \ldots \to N$ for odd t, and take the route $s \to N \to \ldots \to 1$ for even t. Clearly, when N is odd, $\mathcal{R}_v = 1$ and $\mathcal{T}_v = (N+1)/2$ for all $v \in V\backslash s$.

When N is even, we still have $\mathcal{R}_v = 1$ for all $v \in V\backslash s$. However, the delay performance is more complicated. Fig. 1 illustrates the schedule given by LDF or FBA for transmitting the first 10 packets when $N = 4$. Note that at any time, each node uploads and downloads at most one packet.

First, let us show that when N is even, for both LDF and FBA, we have

$$\frac{1}{N}\sum_{v\in V\backslash s}\mathcal{T}_v = \frac{N+1}{2}. \tag{3}$$

Note that $t - N \le |\mathcal{P}_v(t)| \le t$. We have

$$\frac{1}{t}\sum_{\tau=1}^{t-N} T_v(p_\tau) \le \frac{1}{|\mathcal{P}_v(t)|}\sum_{p\in\mathcal{P}_v(t)} T_v(p) \le \frac{1}{t-N}\sum_{\tau=1}^{t} T_v(p_\tau) \tag{4}$$

From the right hand side of (4), we have

$$\begin{aligned} &\tfrac{1}{N}\textstyle\sum_{v\in V\backslash s}\lim_{t\to\infty}\mathcal{T}_v(t)\\ &\le \tfrac{1}{N}\textstyle\sum_{v\in V\backslash s}\lim_{t\to\infty}\tfrac{1}{t-N}\sum_{\tau=1}^{t} T_v(p_\tau)\\ &= (\lim_{t\to\infty}\tfrac{t}{t-N})\cdot(\lim_{t\to\infty}\tfrac{1}{t}\textstyle\sum_{\tau=1}^{t}\tfrac{1}{N}\sum_{v\in V\backslash s} T_v(p_\tau))\\ &= \lim_{t\to\infty}\tfrac{1}{t}\textstyle\sum_{\tau=1}^{t}\tfrac{N+1}{2}\\ &= \tfrac{N+1}{2}.\end{aligned}$$

Similarly, we can show $\frac{1}{N}\sum_{v\in V\backslash s}\mathcal{T}_v \ge (N+1)/2$. Thus, we have proved (3).

However, from the simulation results in Fig. 2 and Fig. 3, we can see FBA incurs much lower variance with respect to the delays at different nodes, and thus has lower worst-case propagation delay. From the numerical results, we have observed that for FBA, when N is even, a half of all the nodes have their $\mathcal{T}_v$ tend to $N/2$ and the other half have their $\mathcal{T}_v$ tend to $N/2+1$ when t is big. Thus, we conjecture that when N is even, the worst-case propagation delay of FBA is

$$\max_v \mathcal{T}_v = \frac{N}{2} + 1. \tag{5}$$

However, to prove such a conjecture rigorously is non-trivial and is not the focus of this paper.

Although LDF and FBA achieve the optimal rates, they both require the packets to "remember" the nodes they have visited in their routes and a high degree of centralized scheduling to reinforce the capacity constraint. Longest Delay First even needs to calculate the propagation delay at each node. Due to these reasons, these algorithms are not easy to implement.

Furthermore, due to the deterministic scheduling of LDF and FBA, delay fairness is hard to be guaranteed — certain nodes always suffer longer delays than others. The deterministic scheduling is also too rigid to be extended to a general network topology with heterogeneous node capacity.

V. A Randomized Approach to Rate-Optimality

In this section, we consider protocols that adopt a random receiver selection scheme, which is extremely flexible in all kinds of networks, suitable for decentralized implementation,

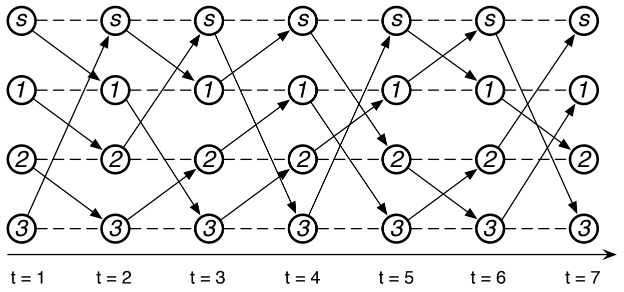

Fig. 4. A trellis graph of 4 nodes. Solid edges model the actual packet transmissions, while the dashed edges model information accumulation in the same node along the time line.

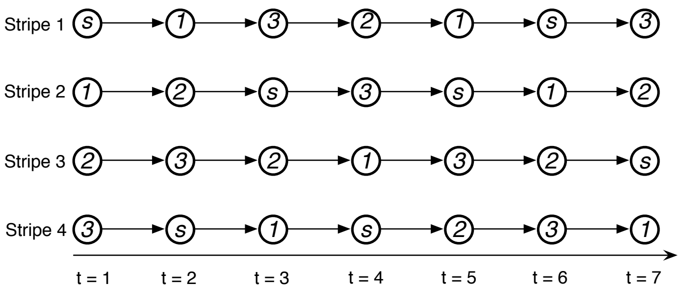

Fig. 5. For random receiver selection in complete graphs, the solid edges of the trellis in Fig. 4 can be decomposed into 4 independent stripes.

while ensuring delay-fairness at all nodes. Consider the case of complete graph first.

- **Random receiver selection.** In each time slot, starting from an arbitrary node, and following an arbitrary order, each node $v \in V$ chooses as its receiver another random node $v_r \in V$ ($v_r \neq v$) that has not yet been chosen, until all $N+1$ nodes have chosen their receivers.

A main result of this paper is the following theorem.

Theorem 1: Assume the network is a complete graph and $U_v = D_v = 1$ for all $v \in V$. Given the random receiver selection, there exists a network code that does not depend on the receiver selection, and can achieve a receiving rate of

$$\mathcal{R}_v^* = 1, \tag{6}$$

which is the optimal receiving rate at node v, for all $v \in V \backslash s$.

Since $\mathcal{R}_v \leq D_v = 1$, $\mathcal{R}_v^* = 1$ is apparently an upper bound on $\mathcal{R}_v$. Now we prove its achievability.

A. Network Coding and the Max-flow Bound

We introduce the following randomized network coding scheme [1] that plays an essential role in proving the achievability of $\mathcal{R}_v^* = 1$.

- **Randomized network coding (RNC).** Assume K packets are to be broadcast to the network from s. At each time t, each sender $v \in V$ linearly encodes all the (coded) packets it has obtained so far using random coefficients in $GF(2^q)$ and transmits the encoded packet.

We consider a trellis graph $G^* = (V^*, E^*)$ constructed from the nodes V, as shown in Fig. 4. For all $v \in V$, let $v_t \in V^*$ represent node v at time t. There is a directed edge of capacity 1 joining v_t and u_{t+1} if node v transmits a packet to node u at time t. To model the information accumulation at the nodes, for each node $v \in V$, we link v_t along the time line with edges of infinite capacity, denoted by the dashed lines. Denote the value of a maximum flow from node s_0 to a node $v_t \in V^*$ by $\text{maxflow}(v_t)$. Then have the following lemma.

Lemma 1: Given the random receiver selection and K packets to be broadcast, randomized network coding achieves the optimal receiving rate for all $v \in V \backslash s$ as $K \to \infty$, and this rate is given by

$$\mathcal{R}_v^* = \lim_{t \to \infty} \frac{1}{t} \text{maxflow}(v_t). \tag{7}$$

*P*roof. Note that G^* is an acyclic graph since each edge in E^* goes from a node at an earlier time to a node at a later time. By the celebrated max-flow bound for multicast in acyclic graphs [2], [3], we have

$$\sum_{\tau=1}^{t} R_v(\tau) \leq \text{maxflow}(v_t), \tag{8}$$

and thus for any $K \geq 1$, we have

$$\mathcal{R}_v = \lim_{t \to \infty} \frac{1}{t} \sum_{\tau=1}^{t} R_v(\tau) \leq \lim_{t \to \infty} \frac{1}{t} \text{maxflow}(v_t). \tag{9}$$

Furthermore, according to [1], such an upper bound is achieved with high probability by applying a random linear code at each node. □

B. The Rate of Latest Packet First

To derive $\mathcal{R}_v^*$, we first derive the rate of an inferior protocol defined as follows:

- **Latest packet first (LPF).** At each time t, the source s transmits a new packet. Each node $v \in V \backslash s$ transmits a packet that it has received in the previous slot.

We then find the difference between $\mathcal{R}_v^{LPF}$ and the max-flow bound $\lim_{t \to \infty} \frac{1}{t} \text{maxflow}(v_t)$ to get $\mathcal{R}_v^*$.

Theorem 2: Assume the network is a complete graph and $U_v = D_v = 1$ for all $v \in V$. For all $v \in V \backslash s$, latest packet first achieves a time average receiving rate of

$$\mathcal{R}_v^{LPF} = \frac{N+1}{2N}, \tag{10}$$

and a time-average propagation delay of

$$\mathcal{T}_v^{LPF} = \frac{2N}{N+1}. \tag{11}$$

To prove this theorem, we need the following lemma:

Lemma 2: Let $\{N(t); t > 0\}$ be a renewal process with renewal epochs $S_1, S_2, \ldots$ and expected inter-renewal time $\mathbf{E}[X] = \overline{X}$. Let $\{R(t); t > 0\}$ be a non-negative randomly varying reward function associated with $\{N(t); t > 0\}$. $R(t)$ depends only on the inter-renewal interval $[S_{N(t)}, S_{N(t)+1})$. Define R_n as the accumulated reward in the nth renewal interval,

$$R_n = \sum_{\tau=S_{n-1}}^{S_n} R(\tau).$$

If $\overline{X} < \infty$ or $\mathbf{E}[R_n] < \infty$, then with probability 1

$$\lim_{t\to\infty} \frac{1}{t} \sum_{\tau=1}^{t} R(\tau) = \frac{\mathbf{E}[R_n]}{\overline{X}}. \tag{12}$$

Proof Sketch. A proof of this theorem for non-arithmatic t is given in [4]. Here we provide a proof sketch for arithmatic t. Please refer to [4] for details. It is not hard to verify for non-negative $R(t)$,

$$\frac{1}{t}\sum_{n=1}^{N(t)} R_n \le \frac{1}{t}\sum_{\tau=1}^{t} R(\tau) \le \frac{1}{t}\sum_{n=1}^{N(t)+1} R_n. \tag{13}$$

The left hand side of (13) can be broken into

$$\frac{\sum_{n=1}^{N(t)} R_n}{t} = \frac{\sum_{n=1}^{N(t)} R_n}{N(t)} \cdot \frac{N(t)}{t} \tag{14}$$

As $t \to \infty$, $N(t) \to \infty$, and thus, $\sum_{n=1}^{N(t)} R_n/N(t) \to \mathbf{E}[R_n]$ w.p. 1 by the strong law of large numbers. Also $N(t)/t \to 1/\overline{X}$ by the strong law for renewal processes. Thus, if $\overline{X} < \infty$ or $\mathbf{E}[R_n] < \infty$, the left hand side of (13) approaches $\mathbf{E}[R_n]/\overline{X}$. Similarly, we can show the right hand side of (13) approaches the same limit and thus prove the theorem. □

Proof of Theorem 2. Let $\mathcal{M}$ denote the set of all the matchings between the transmitting ports and receiving ports of all nodes so that no node is transmitting to itself. It is easy to check the random receiver selection will yield a random matching in $\mathcal{M}$.

Referring to Fig. 4, for latest packet first, the trellis can be decomposed into $N+1$ edge disjoint paths, starting from s, node 1, …, node N, respectively, as shown in Fig. 5. We call each of the edge disjoint paths a stripe. $\mathcal{R}_v$ is the sum of the receiving rate of v in all the stripes. Let $R_v(t) = \sum_{j=1}^{N+1} I_v^j(t)$, where

$$I_v^j(t) = \begin{cases} 1, & \text{if } v \text{ receives a new packet in stripe } j \text{ at } t, \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

we have

$$\mathcal{R}_v = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=1}^{t} \sum_{j=1}^{N+1} I_v^j(\tau) = \sum_{j=1}^{N+1} \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=1}^{t} I_v^j(\tau). \tag{16}$$

It is easy to verify that each stripe forms a random walk with the same transition matrix $\boldsymbol{P} = [p_{ij}]$:

$$p_{ij} = \begin{cases} 1/N, & i, j \in V, i \neq j, \\ 0, & i, j \in V, i = j. \end{cases} \tag{17}$$

Although the $N+1$ stripes start from different nodes, stripes $2, \ldots, N+1$ are delayed renewal processes of stripe 1, which starts from s. Hence, for $j = 2, \ldots, N+1$,

$$\lim_{t\to\infty} \frac{1}{t} \sum_{\tau=1}^{t} I_v^j(\tau) = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=1}^{t} I_v^1(\tau). \tag{18}$$

We can thus get $\mathcal{R}_v$ by determining the receiving rate of node v in stripe 1, and multiplying by $N+1$.

Consider the random walk in stripe 1. We analyze a renewal process defined by the revisits to s. Let T_{vv} denote the time to return to v starting from v for each $v \in V$, and $\pi_v = 1/\overline{T}_{vv}$ be the steady-state fraction of time spent in v. Let q_{vs} denote the probability that starting from s, the random walk in stripe 1 hits v before returning to s. Let R_n denote the accumulation of $I_v^1(t)$ in the nth inter-renewal interval. We have $\mathbf{E}[R_n] = 1 \cdot q_{vs} + 0 \cdot (1 - q_{vs}) = q_{vs}$. Using Lemma 2, we have

$$\lim_{t\to\infty} \frac{1}{t} \sum_{\tau=1}^{t} I_v^1(\tau) = \frac{\mathbf{E}[R_n]}{\overline{T}_{ss}} = \frac{q_{vs}}{\overline{T}_{ss}} \quad \text{w.p. 1.} \tag{19}$$

To derive q_{vs}, we modify the random walk in stripe 1 to a new Markov chain defined by $\boldsymbol{P'} = [p'_{ij}]$. For $i, j \in V$, p'_{ij} satisfies

$$p'_{ij} = \begin{cases} 1/N, & i \neq v \text{ and } i \neq j, \\ 1, & i = v \text{ and } j = s, \\ 0, & \text{otherwise.} \end{cases} \tag{20}$$

The new Markov chain $\boldsymbol{P'}$ is illustrated in Figure 6.

$$s\ v_{i_1}\ v_{i_2}\ \ldots\ s\ \ldots\ v\ s\ \ldots\ v\ s\ \ldots\ s \ldots$$

Fig. 6. An illustration of the modified random walk $\boldsymbol{P'}$ in stripe 1. Note that there is no v in the 1st and 4th inter-renewal interval.

Let π'_v be the steady-state fraction of time spent in v in the new chain and $\overline{T}'_{vv} = 1/\pi'_v$. Consider the renewal process defined by the revisits to s in the new chain. We define a reward function for the new process as

$$I'_v(t) = \begin{cases} 1, & \text{if the new chain is in state } v \text{ at } t, \\ 0, & \text{otherwise.} \end{cases} \tag{21}$$

Let R'_n be the accumulation of the reward $I'_v(t)$ in the nth inter-renewal interval of the new process. Apparently, $\mathbf{E}[R'_n] = 1 \cdot q_{vs} + 0 \cdot (1 - q_{vs}) = q_{vs}$. Using Lemma 2 again, we have

$$\pi'_v = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=1}^{t} I'_v(\tau) = \frac{\mathbf{E}[R'_n]}{\overline{T}'_{ss}} = \frac{q_{vs}}{\overline{T}'_{ss}} \quad \text{w.p. 1.} \tag{22}$$

Combining (19) and (22), we have

$$\lim_{t\to\infty} \frac{1}{t} \sum_{\tau=1}^{t} I_v^1(\tau) = \frac{\pi'_v}{\pi'_s} \cdot \pi_s \quad \text{w.p. 1.} \tag{23}$$

Combining (16), (18), and (23), we obtain

$$\mathcal{R}_v = \frac{\pi'_v}{\pi'_s} \cdot \pi_s \cdot (N+1) \quad \text{w.p. 1.} \tag{24}$$

Clearly, $\pi_s = 1/(N+1)$. π'_s and π'_v satisfy

$$\begin{cases} \pi'_v = (\pi'_o + \pi'_s) \cdot \frac{1}{N} \\ \pi'_s = \pi'_o \cdot \frac{1}{N} + \pi'_v \\ \pi'_s + \pi'_o + \pi'_v = 1, \end{cases} \tag{25}$$

where $\pi'_o = \sum_{i \in V \setminus \{s,v\}} \pi'_i$. Solving (25), we get $\pi'_s = 2N/(N+1)^2$, $\pi'_v = 1/(N+1)$. Substituting into (24), we can obtain (10).

To derive the time-average propagation delay $\mathcal{T}_v$, we notice that starting from state s, we have

$$\begin{aligned} \mathcal{T}_v &= \mathbf{E}[T_{sv} | \text{random walk } \boldsymbol{P} \text{ hits } v \text{ before hitting } s] \\ &= \mathbf{E}[T'_{sv} | \text{random walk } \boldsymbol{P'} \text{ hits } v \text{ before hitting } s] \end{aligned}$$

Assume random walk $\boldsymbol{P'}$ starts from state s. Define Y_1 as $Y_1 := T'_{ss}$ given that random walk $\boldsymbol{P'}$ reaches v before reaching s, and Y_2 as $Y_2 := T'_{ss}$ given that random walk $\boldsymbol{P'}$ reaches s before reaching v. Then $\mathcal{T}_v = \mathbf{E}[Y_1] - 1$.

Let K be the number of "$s \ldots s$" intervals encountered before reaching v, and the sequence of such intervals are denoted $Y_2^{(1)}, Y_2^{(2)}, \ldots, Y_2^{(K)}$. Apparently,

$$T'_{vv} = \sum_{i=1}^{K} Y_2^{(i)} + Y_1. \tag{26}$$

It is not hard to check that K is a stopping time for the sequence of I.I.D. random variables $Y_2^{(1)}, Y_2^{(2)}, \ldots$, and follows the geometric distribution with mean $\mathbf{E}[K] = (1 - q_{vs})/q_{vs} = (\pi'_s - \pi'_v)/\pi'_v$. Applying Wald's equality to (26), we have

$$\begin{aligned} \mathcal{T}_v &= \mathbf{E}[Y_1] - 1 \\ &= \overline{T}'_{vv} - \mathbf{E}[Y_2] \cdot \mathbf{E}[K] - 1 \\ &= \frac{1}{\pi'_v} - \frac{1}{\pi_s^{(V \setminus v)}} \cdot \frac{\pi'_s - \pi'_v}{\pi'_v} - 1, \end{aligned} \tag{27}$$

where

$$\begin{aligned} \mathbf{E}[Y_2] &= \mathbf{E}[T_{ss} | \text{random walk } \boldsymbol{P} \text{ does not visit } v] \\ &= 1/\pi_s^{(V \setminus v)}, \end{aligned} \tag{28}$$

with $\pi_s^{(V \setminus v)}$ being the stationary fraction of time spent in s if v is removed from the network. When the network is a complete graph, $\pi'_v = 1/(N+1)$, $\pi_s^{(V \setminus v)} = 1/N$, and $\pi'_s = 2N/(N+1)^2$. Substituting into (27), we get $\mathcal{T}_v = \frac{2N}{N+1}$. □

C. Deriving $\mathcal{R}_v^*$

Now we are ready to derive the value of $\mathcal{R}_v^* = \lim_{t \to \infty} \frac{1}{t} \text{maxflow}(v_t)$.

Lemma 3: Assume the network is a complete graph and $U_v = D_v = 1$ for all $v \in V$. Referring to the trellis, we have

$$\lim_{t \to \infty} \frac{1}{t} \text{maxflow}(v_t) = 1, \quad \forall v \in V \setminus s. \tag{29}$$

Proof Sketch. Due to the space limit, we provide a sketch of the proof here. Focus on a particular node v. Since $D_v = 1$, we apparently have

$$\lim_{t \to \infty} \frac{1}{t} \text{maxflow}(v_t) \leq 1, \quad \forall v \in V \setminus s. \tag{30}$$

We transform the trellis in Fig. 4 into an equivalent graph in Fig. 7. A max-flow from s_0 to v_t is composed by all the edge-disjoint paths, each from a certain s_{τ_1} to a certain v_{τ_2} with $1 \leq \tau_1 < \tau_2 \leq t$. Two paths are considered *edge-disjoint*, if they do not traverse the same solid edge of capacity 1.

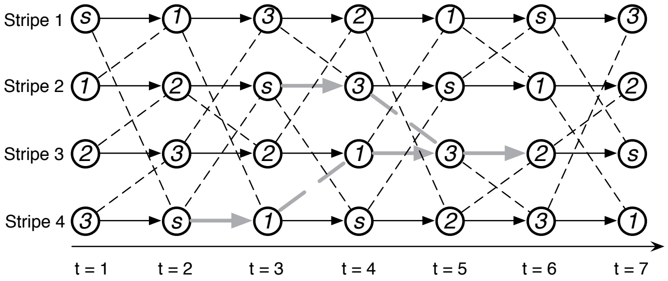

Fig. 7. An equivalent graph of the trellis in Fig. 4. Bold highlighted lines $(s, 3, 3, 2$ and $s, 1, 1, 3)$ are two examples of the cross-stripe paths of the type described in (32).

Such edge-disjoint paths consist of *solid-edge paths* that do not cross stripes, and *cross-stripe paths* that cross stripes via dashed edges. For each s in Fig. 7, let a solid-edge path be the path of solid edges from s to the first v that appears after that s in the same stripe. We call such v a typical v.

The number of typical v's is $\sum_{j=1}^{N+1} \sum_{\tau=1}^{t} I_v^j(\tau)$, as has appeared in (16). To prove (29), we only need to show that the number of edge-disjoint cross-stripe paths $\text{maxflow}(v_t) - \sum_{j=1}^{N+1} \sum_{\tau=1}^{t} I_v^j(\tau)$ satisfies

$$\begin{aligned} &\lim_{t \to \infty} \frac{1}{t} [\text{maxflow}(v_t) - \sum_{j=1}^{N+1} \sum_{\tau=1}^{t} I_v^j(\tau)] \\ \geq\; & 1 - \lim_{t \to \infty} \frac{1}{t} \sum_{j=1}^{N+1} \sum_{\tau=1}^{t} I_v^j(\tau) \\ =\; & 1 - \mathcal{R}_v^{LPF} = \frac{N-1}{2N}. \end{aligned} \tag{31}$$

Thus, we need to find at least one more edge-disjoint path for each of the $(N-1)/2N$ non-typical v's.

For a non-typical v at time $\tau \leq t$, denoted by $v_\tau^{(n)}$, consider a path with $v_\tau^{(n)}$ as the tail and constructed in the following way. Assume $v_\tau^{(n)}$ is preceded by $u_{\tau-1}$ in the same stripe. Let $v_\tau^{(n)}$ traces back to $u_{\tau-1}$, which further traces back to $u_{\tau-2}$, $u_{\tau-3}$, etc., until the path reaches a certain u_{τ_1} $(1 \leq \tau_1 \leq \tau - 2)$, such that in the stripe of u_{τ_1}, u_{τ_1} is immediately preceded by an s, and the $s \ldots s$ interval that contains u_{τ_1} in this stripe does not contain a v. Then a new path

$$p = s_{\tau_1 - 1}, u_{\tau_1}, u_{\tau_1 + 1}, \ldots, u_{\tau-2}, u_{\tau-1}, v_\tau^{(n)} \tag{32}$$

is found to be edge-disjoint with all the solid-edge paths.

Now we show the number of non-typical v's preceded by a u is asymptotically equal to the number of $s \ldots s$ intervals that do not contain a v with the first s succeeded by a u. First, in any interval $[t_1, t_2]$ $(1 \leq t_1 < t_2)$, the number of non-typical v's and the number of $s \ldots s$ intervals that do not contain a v both equal to

$$t_2 - t_1 - \sum_{j=1}^{N+1} \sum_{\tau=t_1}^{t_2} I_v^j(\tau). \tag{33}$$

In complete graphs, the probability that the above $s \ldots s$ interval have the first s succeeded by a $u \notin \{s, v\}$ equals to the probability that the non-typical v is preceded by u, and equals to $1/(N-1)$. During any interval $[t_1, t_2]$ $(1 \leq t_1 < t_2)$, on expectation, we can find the same number of $s_{\tau_1 - 1}, u_{\tau_1}$ and $u_{\tau-1}, v_\tau^{(n)}$ pairs of the above kind. Linking them together via u's and dashed-edges will create a path that is edge-disjoint with all other cross-stripe paths.

By the law of large numbers, as $t \to \infty$, in $[1, t]$, the difference between the number of $s_{\tau_1 - 1}, u_{\tau_1}$ and that of $u_{\tau-1}, v_\tau^{(n)}$ of the above kind divided by t goes to 0. Thus, the total number of edge-disjoint cross-stripe paths created divided by t goes to

$$\lim_{t \to \infty} \frac{1}{t}(t - \sum_{j=1}^{N+1} \sum_{\tau=1}^{t} I_v^j(\tau)) = 1 - \mathcal{R}_v^{LPF}. \tag{34}$$

Hence, we have shown (31) is true and thus proved (29). □

Combining Lemma 1 and Lemma 3, we have proved Theorem 1.

VI. Randomized Protocol for Other Networks

In this section, we analyze the performance of the randomized protocol for a more general class of undirected graphs $G = (V, E)$ that have at least one Hamiltonian cycle.

To better describe the receiver selection algorithm, we define a bipartite graph $G_B = ((V_1, V_2), E_B)$, with V_1, V_2 being copies of V, and denote senders and receivers, respectively. There is an edge between $v_1 \in V_1$ and $u_2 \in V_2$, and an edge between $u_1 \in V_1$ and $v_2 \in V_2$, if and only if there is an edge uv in G.

Note that a bipartite matching in G_B corresponds to a receiver selection schedule in a time slot. Let $\mathcal{M}_p$ denote the set of perfect matchings in G_B. (In a perfect matching, every vertex in G_B is incident to exactly one edge of the matching.) We consider those graphs whose corresponding G_B have at least one *perfect matching*, i.e., $|\mathcal{M}_p| \geq 1$. The random receiver selection is now modified to the following:

- **Random receiver selection.** In each time slot, a pairing relationship between all senders and receivers is chosen so that it corresponds to a random perfect matching in G_B.

Let $G'_B = ((V_1, V_2), E'_B)$, where E'_B is formed by aggregating all $M \in \mathcal{M}_p$. If an edge $u_1 v_2 \in E_B$ appears in multiple perfect matchings, then it counts as multiple edges in E'_B. We state the following fact without proof.

Lemma 4: If $|\mathcal{M}_p| \geq 1$ in G_B, G'_B can be transformed into an equivalent graph $G' = (V, E')$ with node set V and undirected edges E' (multiple edges are possible), so that the number of edges between u_1 and v_2 in G'_B equals to the number of edges between v_1 and u_2 in G'_B, and equals to the number of edges between u and v in G'. This number is denoted by w_{uv}.

Now consider a random walk $\boldsymbol{P} = [p_{ij}]$ $(i, j \in V)$ on G', where a particle at a node v will follow each of its $d(v)$ outgoing edges with probability $1/d(v)$.

Lemma 5: For odd $|V|$, the random walk on G' is irreducible and aperiodic if the original graph G has at least one Hamiltonian cycle.

Proof. If G has a Hamiltonian cycle, this cycle corresponds to a perfect matching in G_B, so that G' also has a Hamiltonian cycle. Hence, G' is strongly connected, and thus the random walk on G' is irreducible. Since the Hamiltonian cycle is of length $|V|$, which is an odd number, G' is non-bipartite. Thus, the random walk on G' is aperiodic (by Lemma 7.12 in [11]). □

Theorem 3: If the original network G has at least one Hamiltonian cycle and $|V|$ is an odd number, then for all $v \in V \backslash s$, latest packet first achieves a time-average receiving rate of

$$\mathcal{R}_v^{LPF} = \frac{d(s) + w_{sv}}{2d(s)}, \tag{35}$$

and a time-average propagation delay of

$$\mathcal{T}_v^{LPF} = \frac{2d(s)}{d(s) + w_{sv}}, \tag{36}$$

where $d(s)$ is the degree of source s in G', and w_{sv} is the number of edges between s and v in G'.

Proof. Since G has a Hamiltonian cycle, by Lemma 5, random walk $\boldsymbol{P}$ on G' is irreducible and aperiodic. Thus, for finite $|V|$, random walk $\boldsymbol{P}$ on G' is ergodic and has a unique stationary distribution $\boldsymbol{\pi} = (\pi_s, \pi_1, \ldots, \pi_N)$.

Similar to the case of complete graph, to derive $\mathcal{R}_v$, we define a modified Markov chain of $\boldsymbol{P}$ as $\boldsymbol{P'} = [p'_{ij}]$. For $i, j \in V$, p'_{ij} satisfies

$$p'_{ij} = \begin{cases} p_{ij}, & i \neq v \text{ and } i \neq j, \\ 1, & i = v \text{ and } j = s, \\ 0, & \text{otherwise.} \end{cases} \tag{37}$$

Let $\boldsymbol{\pi'} = (\pi'_s, \pi'_1, \ldots, \pi'_N)$ be the stationary distribution of the modified chain $\boldsymbol{P'}$. Since each time the sender-receiver pairings corresponding to a perfect matching in $\mathcal{M}_p$, (24) still holds.

It is not hard to check that $\pi_s = d(s)/\sum_v d(v)$ (see Theorem 7.13 [11]). To obtain π'_s and π'_v, we divide node set V into three subsets: $\{v\}$, $\{s\}$ and $\mathcal{O} = V \backslash \{v, s\}$. Let p_{ov}, p_{sv} and p_{os} denote the transition probability from $\mathcal{O}$ to v, from s to v and from $\mathcal{O}$ to s in the modified chain, respectively. Let $\mathcal{N}(v)$ be the neighborhood of v. We have $p_{sv} = w_{sv}/d(s)$, and

$$p_{ov} = \frac{\sum_{u \in \mathcal{N}(v) \cap \mathcal{O}} \pi_u p_{uv}}{\sum_{u \in \mathcal{O}} \pi_u} = \frac{d(v) - w_{sv}}{\sum_{u \in \mathcal{O}} d(u)}, \tag{38}$$

and similarly,

$$p_{os} = \frac{\sum_{u \in \mathcal{N}(s) \cap \mathcal{O}} \pi_u p_{us}}{\sum_{u \in \mathcal{O}} \pi_u} = \frac{d(s) - w_{sv}}{\sum_{u \in \mathcal{O}} d(u)}. \tag{39}$$

π'_s, π'_o, and π'_v satisfy

$$\begin{cases} \pi'_v = \pi'_o p_{ov} + \pi'_s p_{sv} \\ \pi'_s = \pi'_o p_{os} + \pi'_v \\ \pi'_s + \pi'_o + \pi'_v = 1, \end{cases} \tag{40}$$

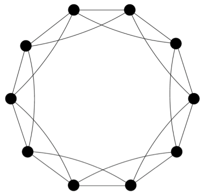

Fig. 8. An illustration of ring lattice. Each node is connected to k other nodes that are within $k/2$ hops away from it on the ring.

which gives

$$\frac{\pi'_v}{\pi'_s} = \frac{p_{ov} + p_{sv}p_{os}}{p_{ov} + p_{os}} = \frac{d(v)d(s) - w_{sv}^2}{(d(v) + d(s) - 2w_{sv})d(s)}. \quad (41)$$

Substituting π'_v/π'_s and π_s into (24), we obtain

$$\mathcal{R}_v = \frac{|V| \cdot (d(s)d(v) - w_{sv}^2)}{(d(v) + d(s) - 2w_{sv}) \sum_v d(v)}. \quad (42)$$

Since G' is equivalent to G'_B, which is formed by aggregating all the perfect matchings in $\mathcal{M}_p$, it is easy to check $\pi_v = \pi_u$, and thus $d(v) = d(u)$, $\forall v, u \in V$. Substituting $d(v) = d(s)$ into (42) proves (45).

Similarly, to derive $\mathcal{T}_v$, we note that (27) still holds. Now $\pi_s^{(V \backslash v)}$ is given by

$$\pi_s^{(V \backslash v)} = \frac{d(s) - w_{sv}}{\sum_{u \in V} d(u) - 2d(v)}. \quad (43)$$

Substituting π'_s, π'_v and $\pi_s^{(V \backslash v)}$ into (27), we can obtain

$$\mathcal{T}_v = \frac{2d(s)d(v) - w_{sv}d(s) - w_{sv}d(v)}{d(s)d(v) - w_{sv}^2}. \quad (44)$$

Substituting $d(v) = d(s)$ into (44) proves (36). □

For this more general type of networks, it is hard to derive the exact rate achieved by randomized network coding. However, we can provide a lower bound on $\mathcal{R}_v^*$ based on Theorem 3.

Corollary 4: If the original network G has at least one Hamiltonian cycle and $|V|$ is an odd number, then for all $v \in V \backslash s$,

$$\mathcal{R}_v^* \geq \frac{d(s) + w_{sv}}{2d(s)}. \quad (45)$$

We demonstrate the use of Theorem 3 and Corollary 4 on a ring lattice illustrated in Fig. 8. Assume the number of nodes in the ring $|V|$ is odd. Each node has a degree of k. Then we have

$$\mathcal{R}_v^* \geq \mathcal{R}_v^{LPF} = \begin{cases} \frac{1}{2} + \frac{1}{2k}, & \text{if } v \text{ is a neighbor of } s, \\ \frac{1}{2}, & \text{otherwise.} \end{cases} \quad (46)$$

VII. Conclusions and Future Work

In this paper, we study the broadcasting problem on an undirected graph, with integer valued node upload and download capacity. We first give two deterministic centralized protocols that achieve the optimal receiving rates, analyze their insufficiency in terms of delay fairness. We proceed to consider a simple randomized decentralized neighbor selection scheme that results in a random matching between all the sending ports and receiving ports of the nodes. We prove that this surprisingly simple protocol can asymptotically achieve the optimal rates at nodes in the long run, for complete and homogeneous networks. Such a proof involves applying randomized network coding at each node. We further extend the results to networks with Hamiltonian cycles and bound the rates of randomized neighbor selection with randomized network coding. We also derive the exact rates achieved by randomized neighbor selection with latest packet first.

Although we have proved the rate-optimality of random neighbor selection, the proof requires applying randomized network coding across an infinite number of packets, which is impractical. Low-complexity network code needs to be cleverly designed to achieve near optimal broadcasting rate. The exact rate achieved by randomized neighbor selection with network coding is yet to be characterized for non-complete graphs with homogeneous node capacity. We conjecture such a rate equals to the optimal rate. Furthermore, it is still an open question to find sufficiently simple decentralized protocols that achieve the optimal rates in heterogeneous networks. One challenging direction for future investigation is to characterize the performance of greedy neighbor selection algorithms for an arbitrary network with heterogeneous node capacity.

References

[1] T. Ho, R. Koetter, M. Medard, D. R. Karger, and M. Effros, "The Benefits of Coding over Routing in a Randomized Setting," in *Proc. of IEEE International Symposium on Information Theory*, 2003.

[2] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung, "Network Information Flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[3] R. Koetter and M. Medard, "An Algebraic Approach to Network Coding," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 782–795, October 2003.

[4] R. G. Gallager, *Discrete Stochastic Processes*. Kluwer Academic Publishers, 1996.

[5] J. Edmonds, "Edge Disjoint Branchings," in *Combinatorial Algorithms*, R. Rustin, Ed. NY: Algorithmics Press, 1972, pp. 91–96.

[6] L. Massoulie, A. Twigg, C. Gkantsidis, and P. Rodriguez, "Randomized Decentralized Broadcasting Algorithms," in *Proc. of IEEE INFOCOM '07*, Anchorage, Alaska, USA, May 2007.

[7] R. W. Yeung, "Avalanche: A Network Coding Analysis," *Communications in Information and Systems*, vol. 7, no. 4, pp. 353–358, 2007.

[8] S. Deb, M. Médard, and C. Choute, "Algebraic Gossip: A Network Coding Approach to Optimal Multiple Rumor Mongering," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2486–2507, June 2006.

[9] D.Mosk-Aoyama and D.Shah, "Information Dissemination via Network Coding," in *Proc. of IEEE International Symposium on Information Theory (ISIT'06)*, Seattle, WA, October 2006.

[10] S. Sanghavi, B. Hajek, and L. Massoulie, "Gossiping with Multiple Messages," in *Proc. of IEEE INFOCOM '07*, Anchorage, Alaska, 2007.

[11] M. Mitzenmacher and E. Upfal, *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.