Stochastic Analysis of Network Coding in Epidemic Routing

Yunfeng Lin, Baochun Li, Ben Liang
Department of Electrical and Computer Engineering
University of Toronto
{ylin, bli}@eecg.toronto.edu, liang@comm.toronto.edu

***Abstract*—Epidemic routing has been proposed to reduce the data transmission delay in disruption tolerant wireless networks, in which data can be replicated along multiple opportunistic paths as different nodes move within each other's communication range. With the advent of network coding, it is intuitive that data can not only be replicated, but also coded, when the transmission opportunity arises. However, will opportunistic communication with network coding perform any better than simple replications? In this paper, we present a stochastic analytical framework to study the performance of epidemic routing using network coding in opportunistic networks, as compared to the use of replication. We analytically show that network coding is superior when bandwidth and node buffers are limited, reflecting more realistic scenarios. Our analytical study is able to provide further insights towards future designs of efficient data communication protocols using network coding. As an example, we propose a priority based coding protocol, with which the destination can decode a high priority subset of the data much earlier than it can decode *any* data without the use of priorities. The correctness of our analytical results has also been confirmed by our extensive simulations.**

I. Introduction

Disruption tolerant wireless networks (DTN), or *opportunistic networks*, represent a class of networks where connections among wireless nodes are not contemporaneous, but intermittent over time. Such networks usually have sparse node densities, with short communication ranges on each node. Connections among nodes may be disrupted due to node mobility, energy-conserving sleep schedules, or environmental interference.

In such networks, an opportunistic link may be temporarily established when a pair of nodes "meet" — when they move into the communication ranges of each other. A possible data propagation path from the source to the destination, referred to as an *opportunistic path*, is composed of multiple opportunistic links, possibly established over time. Clearly, more than one such opportunistic paths may exist. *Epidemic routing* has been proposed [2] to utilize these opportunistic paths to reduce the data transmission delay from a source to a destination, by replicating packets whenever two nodes meet. In essence, epidemic routing replicates data along multiple opportunistic paths from the source to the destination. The delay in delivering a data packet is hence the time to propagate the packet along the opportunistic path with the shortest time.

Randomized network coding [3]–[5] allows intermediate nodes to perform coding operations besides simple replication and forwarding. Using the paradigm of network coding in epidemic routing, a node may transmit a coded packet — as a random linear combination of existing data packets — to another node when the opportunity arises. Intuitively, when replication is used to minimize transmission delay, a node should transmit a packet with the minimum number of replicas in the network, since it is the packet with the longest expected delay. Unfortunately, one does not have precise global knowledge of which packet has the minimum number of replicas in opportunistic networks. When network coding is used, however, a node can transmit *any* coded packet, since all of them can equally contribute to the eventual delivery of all data packets to the destination with high probability.

Though intuition may point to the right direction, how much better does network coding perform as compared to replication, and in what particular scenarios? In this paper, we seek to analytically address this question by presenting a stochastic analysis of network coding in epidemic routing, as compared to the case of replication. Our analysis shows that the use of network coding delivers data with shorter delays when bandwidth is limited, and such advantage may be further magnified when the sizes of node buffers are constrained as well. We show that network coding allows for less buffering capacities than replication, with the same delays required. The correctness of our analytical results has been further confirmed by our extensive simulations.

We believe that insights from our analytical framework are useful towards substantially better designs of new data transmission protocols in disruption tolerant networks. As an example, with network coding, one has to pay the price that any useful data can be decoded only after the destination receives a sufficient number of coded packets and is able to decode all data altogether. That is, the destination may have to wait for a long time before any useful data can be decoded. We propose a priority coding protocol that decodes high priority data much earlier than the time when the original network coding protocol can decode any data. Utilizing our analytical framework, we show that the priority protocol achieves such a goal with low overhead.

The remainder of the paper is organized as follows. We discuss related work in Sec. II. In Sec.III, we present our stochastic analysis of network coding and replication in epidemic routing. In Sec. IV, our analytical framework is extended to study the efficiency of resource usage in both protocols with

This work was supported in part by Bell Canada through its Bell University Laboratories R&D program. A preliminary version of this paper was presented in ACM MobiOpp 2007 [1].

different stopping mechanisms. Sec. V introduces our priority coding protocol, and investigates its design tradeoffs using our analytical framework. In Sec. VI, we present extensive simulation results to show the effectiveness of network coding and to confirm our analytical results. We conclude the paper in Sec. VII.

II. Related Work

A variety of routing protocols have been designed for disruption tolerant networks, based on different sets of assumptions. Some (*e.g.*, [6]) assume *a priori* knowledge on connectivity patterns, or that historical mobility patterns can be used to predict future message delivery probabilities [7]. Others assume control over node mobility [8]. In this paper, we seek a thorough and systematic understanding of the benefits and performance gains when network coding is used in epidemic routing, with neither *a priori* knowledge of network connectivity, nor control over node mobility.

Previous studies have proposed to use erasure coding to address network disruptions in opportunistic networks, with no information of node mobility patterns [9], or with prior knowledge of network topologies [10]. Chen *et al.* [11] further showed a hybrid approach combining erasure coding and replication. Unlike network coding, in such source-based erasure coding approaches, different upstream nodes may transmit duplicates of coded data to the same node, and may unnecessarily consume additional bandwidth.

It has been shown that network coding can improve the throughput in both unicast [12] and broadcast [13] wireless communication, by exploring the broadcast nature of the wireless medium. However, in disruption tolerant networks considered in this paper, a node seldom has more than one neighbors, and such wireless coding opportunities rarely occur.

Deb *et al.* [14] showed that a gossip protocol based on network coding can broadcast multiple messages among nodes with a shorter period of time, as compared to that without network coding, by a logarithmic factor. With the same spirit, the benefit of network coding in wireless broadcast communication has been investigated in [15], [16]. In contrast to their work, we show that network coding can efficiently utilize multiple opportunistic paths in the case of unicast communication in DTNs.

Epidemic routing based on replication has been analytically studied in an extensive set of existing work (*e.g.,* [17]–[21]). We believe that the effects of using network coding in epidemic routing should receive the same levels of rigor and research attention, and the tradeoffs involved with respect to resource consumption and delay should also be carefully studied analytically. Chou *et al.* [5] considered priority encoding in network coding on networks with known topologies. In contrast, our proposed priority coding protocol is designed specifically for opportunistic networks without topology information.

Fluid modeling or differential equations are widely used to model system dynamics, such as in queueing systems [22], P2P networks [23], and DTNs [20], with the attractive advantage of simplicity, as compared to Markov chains. In fact, the system dynamics of one packet in DTN have been modeled with Ordinary Differential Equations (ODEs) in [20] with replication based epidemic routing. By contrast, utilizing ODEs, we study the dynamics of a batch of packets in DTN with both network coding and replication based epidemic routing in this paper.

Zhang *et al.* [24] presented a simulation-based study of the benefit of network coding in opportunistic unicast communication. To our knowledge, this is the closest to our work with respect to research objectives. Nevertheless, our work on stochastic analysis of network coding is analytical in nature, which benefits from the mathematical rigor that is missing in previous work. Inspired by such analysis, we also propose a priority coding protocol to combat the disadvantage of decoding delay in the coding-based protocol. This is an example in which our analytical framework can be used to show how the proposed priority coding protocol is effective with low overhead.

III. Network Coding vs. Replication: an Analytical Framework

A. Network Model

In this paper, we consider unicast communication from a source to a destination in a disruption tolerant network with N wireless nodes, moving within a constrained area. The source has K packets to be transmitted to the destination. A transmission opportunity arises when a pair of nodes "meet," *i.e.*, they are within the communication range of each other. To facilitate the analysis without loss of generality, we assume that when nodes i and j meet, the transmission opportunity is only sufficient to completely transmit one data packet. It is straightforward to extend this to the general case where an arbitrary number of data packets can be delivered when the opportunity arises, as sketched at the end of Sec. III-B. With respect to the buffering capacities, while the source and the destination are able to accommodate all K packets, we assume that the buffer on each of the intermediate relay nodes is only able to hold B packets, where $1 \leq B \leq K$.

We assume that the time between two consecutive transmission opportunities (when nodes meet) is exponentially distributed with a rate of λ. In the literature, the majority of previous work makes such an assumption, either explicitly [19], [20] or implicitly [17], [18]. Although measurement-based studies (*e.g.*, [25]) have shown that such inter-meeting times may follow heavy-tail distributions in some applications, more recent studies have shown that the exponential distribution is in fact more prevalent both in theory and in many practical systems [26], [27]. Therefore, we opt for more mathematical tractable models in our analysis, and believe that insights obtained from our analysis are also useful under other realistic mobility models. With a similar preference for mathematical tractability, we assume there does not exist background traffic beyond the unicast communication under consideration, and leave the more general case with background traffic to our future work.

B. Epidemic Routing with Network Coding

We are now ready to develop an analytical model for network coding in opportunistic networks with epidemic routing. The following baseline protocol will be analyzed. When two nodes a and b meet, they transmit coded packets to each other. A coded packet x is a linear combination of the K source packets $E_1, \ldots, E_K$ in the form: $x = \sum_{i=1}^{K} \alpha_i E_i$, where α_i are coding coefficients. Suppose that node a holds m coded packets in its buffer, where $1 \leq m \leq B$. Node a encodes all coded packets in its buffer, namely $x_1, \ldots, x_m$, to generate a coded packet x_a:

$$x_a = \sum_{i=1}^{m} \beta_i x_i, \tag{1}$$

where all multiplication and addition operations are defined on a Galois field (such as GF(2^8) when the operations are performed on each byte), and β_i is randomly chosen from the field. It is easy to see that x_a is also a linear combination of the K original packets, and the coding coefficients can be derived. Node a then transmits x_a along with its coding coefficients to node b. When node b receives x_a, it stores x_a in its buffer if space is available. Otherwise, node b encodes x_a with each packet in its buffer as follows:

$$x'_i = x'_i + \alpha_i x_a, \tag{2}$$

where x'_i represents the ith coded packet in the buffer of node b, and α_i is randomly chosen from the Galois field.

The destination obtains a coded packet when it meets another node, and attempts the decoding process to retrieve K source packets as long as K coded packets have been collected. Because the coding coefficients and the coded packet are known, each coded packet represents a linear equation with the K source packets as unknown variables. Decoding the K source packets is equivalent to solving the linear system composed of K coded packets. The *decoding matrix* represents the coefficient matrix of such a linear system. When the rank of the decoding matrix is K, the linear system can be solved and the K source packets are decoded. Otherwise, there exists linear dependence among the K coded packets, and the node will continue to obtain more coded packets until decoding is successful.

With such a protocol using network coding, the ultimate objective of our stochastic analysis is to compute the delivery delay of all K packets from the source to the destination. If there are more nodes with coded packets in their buffers, the destination has a higher probability to obtain a useful coded packet from a transmission opportunity with another node, and proceeds towards the decoding of all K packets. Hence, to compute the delivery delay of all K packets from the source to the destination, we first compute the packet distribution on relay nodes.

Recall that B denotes the maximum relay buffer size. We classify relay nodes in the network into $B+1$ types using the number of packets a node has: nodes of type v_i each has i packets, where $0 \leq i \leq B$. For clarity, a node of type v_i is henceforth referred to as v_i. We examine the transmission opportunity when two relay nodes meet. We say that one node can transmit an *innovative* coded packet to another node, if the coded packet it transmits can increase the rank of the decoding matrix on the other node. Clearly, v_i can transmit an innovative coded packet to v_j, if $i > j$, assuming the coded packets in the buffers are linearly independent with high probability. We make the following important assumption in the analysis: *if $i \geq 1$ and $j < K$, v_i can transmit an innovative coded packet to v_j with high probability, even when $i \leq j$.* In the case of abundant buffers, Deb *et al.* [14] have shown that the probability that a coded packet is useful to another node is $1 - 1/q$, where q is the size of the Galois field to generate random coding coefficients. In practice, q is usually sufficiently large such that $1-1/q$ is very close to 1[1]. Although the relay buffer is limited in our protocol, we will see that our stochastic analysis based on such an assumption is still very close to the simulation result in Sec. VI.

We then define the *network state*, the packet distribution on relay nodes by a B-tuple $\{X_1(t), \ldots, X_B(t)\}$, where $X_i(t)$ denotes the number of type v_i nodes in the network. We further use $X_0(t)$ to represent the number of type v_0 nodes and its value is $N - \sum_{i=1}^{B} X_i(t)$. In the following, we characterize the network state using ODEs.

Let $D_i(t)$ denote the receiving rate of v_i, defined as the expected number of innovative coded packets received in a unit time interval, for $0 \leq i \leq B$. We further use $D_{B+1}(t) \ldots D_K(t)$ to denote the receiving rate of the destination, when it has obtained $B+1, \ldots, K$ packets, respectively. For v_0, it can receive an innovative coded packet from any relay node with at least one coded packet, namely v_j, where $1 \leq j \leq B$, and the source node with probability 1. For v_i, it can receive an innovative coded packet from v_j with high probability, if $j \geq 1$ as discussed previously, and from the source with probability 1. However, a node cannot transmit an innovative packet to itself, and such quantity should be excluded in the equation. This is the reason leading to the difference between the first and second equation in (3). Similar arguments also apply to the receiving rates of the destination. Hence, we have

$$\begin{aligned} D_0(t) &= \lambda(\sum_{j=1}^{B} X_j(t) + 1), \\ D_i(t) &= \lambda(\sum_{j=1}^{B} X_j(t)), \quad \text{for} \quad i = 1, 2, \ldots, K-1, \\ D_K(t) &= 0, \end{aligned} \tag{3}$$

where $D_0(t), \ldots, D_B(t)$ are applicable for both relay nodes and the destination, whereas $D_{B+1}(t), \ldots, D_K(t)$ are applicable for only the destination since relay nodes can hold at most B packets and all packets in the relay buffer are linearly independent with high probability. Furthermore, $D_B(t)$ is for the destination in (3), whereas $D_B(t) = 0$ for the relay nodes as the relay buffer size is B.

Next, we consider the varying rate of $X_i(t)$ within a short time interval, which is composed of two parts. First,

[1] A byte is usually used to store a coding coefficient for the tradeoff among ease of implementation, overhead, and sufficiently linear independence among coded packets. Hence, q is $2^8 = 256$.

$D_{i-1}(t)X_{i-1}(t)$ number of v_{i-1} becomes v_i since they can obtain one innovative coded packet with high probability. Second, $D_i(t)X_i(t)$ number of v_i becomes v_{i+1} since they can also acquire one innovative coded packet with high probability. Therefore, we use the following ODEs to compute $X_i(t)$:

$$\frac{dX_i}{dt} = D_{i-1}(t)X_{i-1}(t) - D_i(t)X_i(t), \quad \text{for} \quad i = 1, \ldots, B-1,$$
$$\frac{dX_B}{dt} = D_{B-1}(t)X_{B-1}(t), \tag{4}$$

where $D_i(t)$ is computed in (3) as a function of $X_i(t)$. The above ODEs can be solved with the initial value $X_i(t) = 0$ for $i = 1, \ldots, B$.

We proceed to compute the distribution of the delivery delay from the time that the source begins transmitting data to the time that the destination obtains all K packets. We use the random variable T_i to denote the time that the destination obtains i packets, for $i = 1, \ldots, K$. Hence, the delivery delay for all K packets is T_K. Let $F_i(t)$ be the Cumulative Distribution Function (CDF) of T_i. We derive $F_i(t)$ with ODEs by computing the derivative of $F_i(t)$ over t. In particular, to derive $F_K(t)$, *i.e.* $\Pr(T_K < t)$ with ODEs, we compute the value change of $\Pr(T_K > t)$ within a small time interval $[t, t+\delta t]$. Hence, we can compute the CDF $F_K(t)$ of the delivery delay of K packets by solving the following ODEs, where details of the derivation are presented in Appendix A:

$$\frac{dF_1}{dt} = D_0(t)(1 - F_1(t)),$$
$$\frac{dF_i}{dt} = D_{i-1}(t)(F_{i-1}(t) - F_i(t)), \quad \text{for} \quad i = 2, \ldots, K. \tag{5}$$

The initial values of the above ODEs are $F_i(0) = 0$, for $i = 1, \ldots, K$, and $D_i(t)$ is given in (3) by solving (4).

In the above analytical framework, we assume that the bandwidth when two nodes meet is only sufficient to transmit one packet. Here, we briefly outline how to extend the basic analytical framework to the more general case where at most b packets can be transmitted when two nodes meet, for the network coding case. The extension for the replication case is similar. Because such an extension on the basic framework is straightforward but complicated, and does not offer significant new insight, we omit further details in the later parts of this paper.

We use the same notation $X_i(t)$ to denote the number of nodes with i coded packets in their buffers. Similarly, we focus on the changing rate of $x_i(t)$, which can be characterized by the following equation:

$$\frac{dX_i}{dt} = \lambda(X_{i-b}\sum_{j=b}^{B} X_j + \sum_{j=1}^{b-1} X_{i-b+j}X_{b-j} - X_i \sum_{j=1}^{B} X_j) \tag{6}$$

(6) holds because once v_{i-b} meets v_j with $j \geq b$, this v_{i-b} will become v_i with b innovative coded packets. Therefore, $X_i(t)$ increases by $\lambda X_{i-b}\sum_{j=b}^{B} X_j$. Furthermore, when v_{i-b+j} meets v_{b-j}, where $1 \leq j \leq b-1$, it becomes v_i by obtaining $b-j$ innovative coded packets. Hence, $X_i(t)$ increases by $\lambda \sum_{j=1}^{b-1} X_{i-b+j}X_{b-j}$. Finally, if v_i meets any node with a non-empty buffer, it will become v_l, where $l > i$, by retrieving innovative coded packets from the node that it meets. Therefore, $X_i(t)$ decreases by $\lambda X_i \sum_{j=1}^{B} X_j$. Note in the above argument, we ignore the tedious details that the source may transmit innovative packets and the node cannot transmit to itself for presentation clarity. Based on (6), we are able to construct the analytical framework in the similar way as the above basic analytical framework.

C. Epidemic Routing with Replication

We now present a stochastic analytical model for epidemic routing with replication, to serve as a comparison with our analytical results in the case of network coding.

With epidemic routing using replication, when two nodes a and b meet, node a knows the set of packets in node b and vice versa, through the exchange of packet identifiers. Let S_a and S_b denote the set of packets on node a and b, respectively. In the following, we describe the protocol for only node a since the protocol for node b is identical. Node a chooses one packet in the set $S_a - S_b$ to transmit to node b such that the packet transmitted to node b is always new to node b. If $S_a - S_b$ is empty, node a will miss this transmission opportunity.

We examine three policies in selecting which packet from $S_a - S_b$ is to be transmitted. First, in the *random* policy, node a chooses a packet with the same probability for each packet in $S_a - S_b$. Second, in the *local rarest* policy, node a uses a counter for each packet in the buffer to record how many times that each packet has been transmitted and chooses the packet with the smallest counter. Third, in the *global rarest* policy, we assume that an oracle maintains global counters for all K packets, *i.e.*, the number of copies of each packet in the network. Node a chooses the packet with the smallest counter to transmit. It is clear that the last two policies try to maintain an even distribution of the copies of the K different packets in the network. Although the global rarest policy is impractical, by comparing it with the other two policies, we will have a clearer understanding on the difference between simulation and analytical results, as we will show in Sec. VI.

Upon receiving a packet P_a from node a, node b inserts P_a into its buffer if the buffer is not full. If the buffer is already full, node b uses P_a to replace a random packet in its buffer in the random policy. In the local or global rarest policy, node b compares the counter of P_a with the counter of P_b which is the packet with the largest counter in node b's buffer, and drops the packet among P_a and P_b with the larger counter.

We proceed to study the delivery delay of the above protocol based on replication. Similar to our analysis in the case of network coding, we first compute the packet distribution on relay nodes. We similarly classify relay nodes in the network by $B+1$ types, denoted by v_i, for $i = 0, \ldots, B$. We make the following assumption in our analysis. *The i packets on v_i are uniformly distributed among the K original packets.* This assumption is reasonable if the global rarest policy are employed since it maintains close to even proportions of K packets in the network. We will show the accuracy of this assumption on all three policies in our simulation-based studies. We then examine the probability $\Pr(i, j)$ that

v_i obtains a new packet from v_j under such an assumption. First, it is easy to see that, if $i < j$, v_i can always obtain a new packet from v_j. Second, if $i \geq j$, v_i cannot obtain a new packet from v_j only if v_i contains all packets on v_j, which has the probability $\binom{i}{j}/\binom{K}{j}$ under the assumption of uniform packet distribution. Hence, we have $\Pr(i,j) = 1 - \binom{i}{j}/\binom{K}{j}$ in such a case. In summary, we have

$$\Pr(i,j) = \begin{cases} 1 & \text{if } i < j, \\ 1 - \binom{i}{j}/\binom{K}{j} & \text{if } i \geq j. \end{cases} \tag{7}$$

We note that similar analysis has been applied to BitTorrent like P2P file sharing systems (*e.g.*, [28]).

Again, a B-tuple $\{X_1(t), \ldots, X_B(t)\}$ is used to represent the network state at time t. Let $D_i(t)$ denote the receiving rate of v_i, for $1 \leq i \leq B$. We further use $D_{B+1}(t) \ldots D_K(t)$ to denote the receiving rate of the destination, when it has obtained $B+1, \ldots, K$ packets, respectively. For v_0, it can receive a new packet from any relay node with at least one packet, namely v_j where $1 \leq j \leq B$, and from the source with probability 1. For v_i, it can receive new packets from v_j with probability $\Pr(i,j)$ and from the source with probability 1. However, a node with i packets can not transmit an innovative packet to itself, and such quantity should be excluded in the equation, causing the term $-\Pr(i,i)$ in the second equation of (8). Similar arguments also apply to the receiving rates of the destination. Hence, we have

$$\begin{aligned} &D_0(t) = \lambda(\sum_{j=1}^{B} X_j(t) + 1), \\ &D_i(t) = \lambda(\sum_{j=1}^{B} X_j(t)\Pr(i,j) + 1 - \Pr(i,i)), \\ &\qquad \text{for } i = 1, \ldots, K-1, \\ &D_K(t) = 0, \end{aligned} \tag{8}$$

where $\Pr(i,j)$ is computed in (7). $D_0(t), \ldots, D_B(t)$ are applicable for both relay nodes and the destination, whereas $D_{B+1}(t), \ldots, D_K(t)$ are applicable for only the destination since relay nodes can hold at most B packets. Furthermore, $D_B(t)$ is for the destination in (3), whereas $D_B(t) = 0$ for the relay nodes as the relay buffer size is B.

Finally, we can use the same set of ODEs in (4) and (5) to obtain $X_i(t)$ and $F_K(t)$, the CDF of the delivery delay of all K packets, by replacing $D_i(t)$ for the coding based protocol in them with the values computed in (8) for the replication based protocol.

IV. Stopping Mechanisms and Resource Usage Analysis

We proceed to analyze the protocol resource usage under different mechanisms to stop the packet transmissions.

A. Reactive Stopping Mechanisms

We first study the network resource usage under the recovery schemes proposed in [17]. For replication based protocol, if a relay node transmits a packet, *e.g.* packet i, to the destination, this relay node can remove packet i from its buffer to save buffer space since packet i is successfully delivered to the destination. In addition, this relay node can record the delivery information by carrying an "ACK" for packet i in its buffer to keep it from receiving packet i from other relay nodes again. Such scheme is referred to as IMMUNE in [17]. A more efficient scheme, VACCINE [17], is to propagate ACKs among relay nodes rather than only from the destination to relay nodes as in IMMUNE. Since these mechanisms are activated only after the destination has received data, we refer them to as *reactive stopping mechanisms*.

Similar recovery schemes can also be designed for network coding based protocol. Unlike the replication based protocol, where K different ACKs are used to acknowledge the K different data packets, only one type of ACKs exist in the network coding based protocol for all K packets. The destination generates such ACKs after it has decoded all K source packets.

We focus on two metrics of network resource usage: the buffer consumption of relays nodes and the number of transmissions made by relay nodes. The network resource usage are important if multiple sessions of network traffic compete for limited resource. Furthermore, the number of transmissions by relay nodes implies their energy consumption and is critical for mobile nodes with limited energy.

1) Epidemic Routing with Network Coding: Before the analysis for resource usage for the coding based protocol, we first compute the probability $S_i(t)$ that the destination receives i packets at time t from the CDFs $F_i(t)$ and $F_{i+1}(t)$ of T_i and T_{i+1}, which are derived in (5). We have

$$\begin{aligned} S_i(t) &= \Pr(T_i < t, T_{i+1} > t) \\ &= \Pr(T_i < t) - \Pr(T_i < t, T_{i+1} < t) \\ &= \Pr(T_i < t) - \Pr(T_{i+1} < t) \\ &= F_i(t) - F_{i+1}(t) \end{aligned} \tag{9}$$

for $i = 1, \ldots, K-1$, where the second equality holds because the event $\{T_{i+1} < t\}$ implies the event $\{T_i < t\}$ due to $T_i \leq T_{i+1}$. Similarly, we have

$$\begin{aligned} S_0(t) &= 1 - F_1(t), \\ S_K(t) &= F_K(t). \end{aligned} \tag{10}$$

We first consider the case for VACCINE, where the ACKs are propagated among all relay nodes. We use v_i to denote the relay nodes with i coded packets in their buffer as in Sec. III-B, and v_R to represent the relay nodes that have received the ACK. Let $\{Y_1(t), \ldots, Y_B(t), Y_R(t)\}$ be the network state, where $Y_i(t)$ denotes the number of v_i, for $i = 1, \ldots, B$, and $Y_R(t)$ denotes the number of v_R. We then have the following set of ODEs to describe the dynamics of the network state:

$$\frac{dY_1}{dt} = \lambda Y_0(\sum_{j=1}^{B} Y_j + 1) - \lambda Y_1(\sum_{j=1}^{B} Y_j + Y_R + S_K), \tag{11}$$

$$\begin{aligned} &\frac{dY_i}{dt} = \lambda Y_{i-1}(\sum_{j=1}^{B} Y_j) - \lambda Y_i(\sum_{j=1}^{B} Y_j + Y_R + S_K), \\ &\qquad \text{for} \quad i = 2, \ldots, B-1 \end{aligned} \tag{12}$$

$$\frac{dY_B}{dt} = \lambda Y_{B-1}(\sum_{j=1}^{B} Y_j) - \lambda Y_B(Y_R + S_K), \quad (13)$$

$$\frac{dY_R}{dt} = \lambda(\sum_{j=0}^{B} Y_j)(Y_R + S_K), \quad (14)$$

where $S_K(t)$ is derived in (10), and $Y_0(t) = N - Y_R(t) - \sum_{i=1}^{B} Y_i(t)$ represents the number of relay nodes with no coded packets and the ACK. (12) holds because once v_{i-1} meets v_j with $j \geq 1$ or the source, and excluding itself, this v_{i-1} will become v_i with a new innovative coded packet. Therefore, $Y_i(t)$ increases by $\lambda Y_{i-1}(t)(\sum_{j=1}^{B} Y_j)$. On the other hand, if v_i meets v_j with $j \geq 1$, the source, and excluding itself, it becomes v_{i+1}. Furthermore, if v_i meets v_R or the destination with K packets, it will become v_R. Hence, $Y_i(t)$ decreases by $\lambda Y_i(\sum_{j=1}^{B} Y_j + Y_R + S_K)$. Similar argument applies for (11), (13), and (14). The initial values for the above ODEs are $Y_i(0) = 0$ and $Y_R(0) = 0$, for $i = 1, \ldots, B$.

For IMMUNE, a similar set of ODEs can be constructed to characterize the network state as follows:

$$\frac{dY_1}{dt} = \lambda Y_0(\sum_{j=1}^{B} Y_j + 1) - \lambda Y_1(\sum_{j=1}^{B} Y_j + S_K),$$

$$\frac{dY_i}{dt} = \lambda Y_{i-1}(\sum_{j=1}^{B} Y_j) - \lambda Y_i(\sum_{j=1}^{B} Y_j + S_K),$$

$$\text{for} \quad i = 2, \ldots, B-1$$

$$\frac{dY_B}{dt} = \lambda Y_{B-1}(\sum_{j=1}^{B} Y_j) - \lambda Y_B S_K,$$

$$\frac{dY_R}{dt} = \lambda(\sum_{j=0}^{B} Y_j) S_K, \quad (15)$$

with the same initial values as in (11), (12), (13), and (14). Comparing the two sets of ODEs to describe VACCINE and IMMUNE, we can see the factor $Y_R + S_K$ in the right of equations is reduced to S_K to reflect the fact that only the destination can propagate ACKs.

By solving $Y_i(t)$ from the above ODEs, we have the buffer consumption of the network: the expected number of total coded packets $C(t)$ among all relay buffers is

$$C(t) = \sum_{i=1}^{B} Y_i(t) \cdot i \quad (16)$$

since v_i holds i coded packets in its buffer.

Furthermore, the number of relay transmissions in the network can also be computed from $Y_i(t)$. Since any v_i for $i > 1$ will transmit as long as they meet another node that has not recovered, and the meeting rate of relay nodes is λ. Hence, the expected number of relay transmissions is

$$D(t) = \lambda(\sum_{i=1}^{B} Y_i(t))(N - Y_R(t)) \quad (17)$$

2) Epidemic Routing with Replication: Similarly, the probability $S_i(t)$ that the destination has received i packets can be computed with (9) and (10), except where $F_i(t)$ is replaced with the delay distribution of the replication based protocol derived at the end of Sec. III-C: the replication version of (5).

We first study VACCINE. Since the size of ACKs is much smaller than data packets, we assume that an arbitrary number of ACKs can be transmitted when two nodes meet. We use $\{Z_1(t), \ldots, Z_K(t)\}$ to denote the state of ACKs in the network, where $Z_i(t)$ represents the number of relay nodes with i ACKs. When two relay nodes, *e.g.* node a and node b meet, suppose they have c_a ACKs and c_b ACKs in their buffer, denoted by A_a and A_b, respectively. It is easy to see that $A_a \subset A_b$, if $c_a < c_b$, because all ACK are originated from the same node, the destination. Therefore, after exchanging of ACKs, node b will also have c_a ACKs. When a node meets the destination at time t, its ACKs will be increased to i with probability $S_i(t)$, the probability that the destination has obtained i packets at time t. Hence, we have the following set of ODEs to characterize the dynamics of ACKs in the network:

$$\frac{dZ_i}{dt} = \lambda \sum_{j=0}^{i-1} Z_j(Z_i + S_i) - \lambda Z_i \sum_{j=i+1}^{K} (Z_j + S_j),$$

$$\text{for} \quad i = 1, \ldots, K-1, \quad (18)$$

$$\frac{dZ_K}{dt} = \lambda \sum_{j=0}^{K-1} Z_j(Z_K + S_K), \quad (19)$$

(18) holds since $Z_i(t)$ increases if a relay node with less than i ACKs meets a relay node or the destination with i ACKs. Furthermore, $Z_i(t)$ decreases if a relay node with i ACKs meets a node with more than i ACKs. Similar justification applies for (19).

For IMMUNE, similarly, the set of ODEs to characterize the dynamics of ACKs is as follows:

$$\frac{dZ_i}{dt} = \lambda \sum_{j=0}^{i-1} Z_j S_i - \lambda Z_i \sum_{j=i+1}^{K} S_j, \quad \text{for} \quad i = 1, \ldots, K-1, \quad (20)$$

$$\frac{dZ_K}{dt} = \lambda \sum_{j=0}^{K-1} Z_j S_K, \quad (21)$$

(20) holds since $Z_i(t)$ increases if a relay node with less than i ACKs meets the destination and the destination has i ACKs at time t. Furthermore, $Z_i(t)$ decreases if a relay node with i ACKs meets the destination and the destination has more than i ACKs at time t.

We proceed to derive the expected number of data packets in relay buffers at time t, using $X_i(t)$ (derived at the end of Sec. III-C, the replication version of (4), *under the assumption that no ACKs are propagating*) and $Z_i(t)$ for both VACCINE and IMMUNE. We make two modeling assumptions here. First, we assume that data packet transmissions are not significantly affected by the ACKs. In reality, ACKs do affect data packet transmissions since a node that has obtained ACK for data packet a will not involve in replicating packet a. This assumption is sufficiently accurate for IMMUNE, but is only an approximation for VACCINE, because the ACK propagation for IMMUNE is much slower than VACCINE. We confirm the accuracy of such an assumption for different mechanisms via simulations in Sec. VI-B. Second, we assume the *virtual* data packets, the data packets in the buffer and the data packets that have been removed by ACKs, are transmitted independent of ACK transmissions on each relay node, justified by the fact that the data packets may arrive at the destination in any order, and the destination generates ACKs for different data packets in any order as well. Hence, for a particular node a with i data packets (including the packets that have been removed by ACKs) and j ACKs, since ACKs and data packets are assumed to be independent on a particular node, each of the i data packets has j/K probability to be acknowledged and removed from the buffer. Hence, the expected number of data packets that have been removed is ij/K, and the expected number of data packets remained in buffer is $i(1-j/K)$. Given a node with i data packets, it has $Z_j(t)/N$ probability with j ACKs. Therefore, the expected number of data packets in its buffer is $\sum_{j=0}^{K} \frac{Z_j(t)}{N} i(1-j/K)$. Hence, the total expected number of data packets $C(t)$ in all relay buffers is

$$C(t) = \sum_{i=0}^{B} X_i(t) (\sum_{j=0}^{K} \frac{Z_j(t)}{N} i(1 - \frac{j}{K})) \tag{22}$$

Finally, the more involved analysis for the amount of relay transmissions is presented in Appendix B.

B. Proactive Stopping Mechanism for Network Coding Based Protocol

In the network coding based protocol described so far, two nodes exchange coded packets whenever they meet until an ACK from the destination indicating all K packets have been received. Such protocol may have many transmissions and consume a large amount of energy. In this section, we propose an efficient variant of the network coding based protocol that has significantly less amount of transmissions while increasing packet delivery delay only slightly. Our protocol is based on a *proactive stopping mechanism*, where counters are used to stop the relay transmissions before the destination decodes all data.

Our design is motivated by the following fundamental question: what is the minimal number of transmissions that should be made by the source and the relays to achieve the minimal delivery delay? To deliver K data packets from the source to the destination, it is easy to see from the information-theoretical perspective that the source needs to transmit at least K coded packets to either relay nodes or directly to the destination. Furthermore, to achieve the minimal delivery delay, the destination should decode all packets after obtaining K coded packets. Hence, the minimal number of transmissions made by relay nodes should disseminate and mix the K coded packets from the source such that the destination can decode all packets by obtaining K coded packets from *any* K relay nodes with high probability.

In the proposed protocol, the source transmits slightly more than K *coded* packets into the network such that these coded packets are sufficient to decode the original packets with high probability. All these coded packets are referred to as *pseudo source packets*. Each pseudo source packet is then disseminated to L random nodes in the network in the same spirit as "Binary Spraying" in [21]. Note that we also encode them together during the dissemination whenever possible. Spyropoulos *et al.* [21] have shown that "Binary Spraying" is the optimal spraying method with the minimal packet delivery delay under a homogeneous random mobility model such as ours. By adjusting L, referred to as the *maximal spray counter* hereafter, we can tune the tradeoff between the number of relay transmissions and packet delivery delay. The question is whether there is a critical threshold such that the protocol performance will degrade dramatically if L is below it. We delay such analysis after we describe the details of the protocol.

The protocol proceeds as follows. The source maintains a counter S with an initial value K' slightly larger than K. When the source meets a relay node, if $S > 0$, it generates a coded packet from its buffer (a pseudo source packet) using the algorithm presented in Sec. III-B, and transmits the packet to the relay node. We order the pseudo source packet from the source with indices $1, \ldots, K'$. Each packet from the source carries its index i and spray counter l, which is initialized to the maximal spray counter L. The source decreases S by one after each transmission to a relay node and stops transmissions if $S = 0$.

The relay nodes implement the "Binary Spraying" protocol for each pseudo source packet, while encoding them together whenever possible. Every relay node, *e.g.*, node a, keeps a list of tuples: $\langle i, l \rangle$, where i and l denote the index of the pseudo source packet and the value of its spray counter. Such lists are referred to as *spray list* and are empty initially. When node a meets b, it checks the spray lists in both nodes. If node a finds a tuple $\langle i, l \rangle$ with $l \geq 2$ and there is no tuple with i as the first element in node b, node a transmits a coded packet to node b; otherwise, node a misses the transfer opportunity. If node a decides to transmit, it generates a coded packet, using the algorithm in Sec. III-B, and sets the packet index i and the new spray counter $\lfloor l/2 \rfloor$ to be carried in the coded packet. Node a then updates its tuple with $\langle i, \lceil l/2 \rceil \rangle$. Upon receiving a coded packet, node b stores or encodes the coded packet with the algorithm in Sec. III-B. Furthermore, node b inserts a new tuple into its list: $\langle i, l \rangle$, where i and l are the packet index and spray counter carried in the incoming coded packet, respectively. Note that the source and relay nodes always generate a coded packet and transmit to the destination regardless of the counter S or the spray list.

To analyze the amount of transmissions generated by this protocol, we first state the following obvious fact: under the homogeneous random mobility model, the L nodes selected by the "Binary Spray" protocol are uniformly distributed among the N relay nodes, which is easy to see since each node has the same probability to meet another node. We then characterize the asymptotical optimal bound of L. Since the proactive stopping mechanism is more useful when the amount of data to be transmitted is large, we assume $K = \Theta(N)$ throughout this section. However, we emphasize that this assumption is only relevant in the proactive stopping mechanism, and there is no requirement on the relation between K and N in all other modeling parts of this paper.

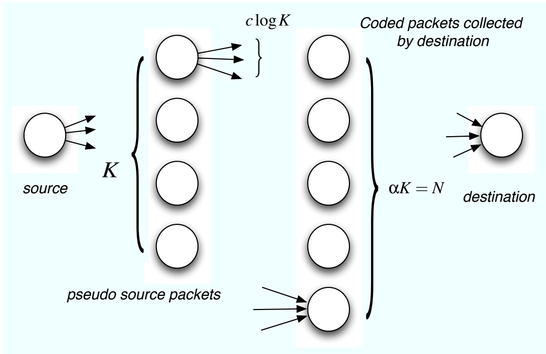


Fig. 1. The network-flow formulation in Theorem 1, where c and α are constants.

Theorem 1: if each node has buffer size K, the maximal spray counter L should be $\Theta(\log K)$ in order for the destination to decode all K source packets with K coded packets with high probability.

Proof: We reduce our problem to the problem studied in [29] by a network-flow formulation as shown in Fig. 1. The slightly more than K coded packets from the source can be equivalently considered as K linearly independent pseudo source packets. With the coding based protocol, the coded packets in relay nodes are the random linear combination of the K pseudo source packets. Moreover, as we shown previously, the information of a pseudo source packet is disseminated to L uniformly random relay nodes by the transmissions corresponding to the spray counter indexed by this pseudo source packet. Furthermore, because each relay node has buffer size K, it is not hard to see that the transmissions of different pseudo source packets to their L relay nodes are independent. Hence, Theorem 1 and 2 in [29] apply here. They show that a source packet needs to be disseminated to only $\Theta(\log K)$ random nodes in the network in order for the destination to decode all source packets with *any* K coded packets from any K nodes with high probability. Hence, L should be $\Theta(\log K)$. ■

We then have the following corollary and lemma on the amount of transmissions made by the relay nodes. Therefore, we conclude that the new protocol is asymptotically more efficient than the original protocol in transmissions.

Corollary 1: In the efficient protocol with proactive stopping, the relay nodes transmit $\Theta(\log K)$ data packets.

Proof: There are K pseudo source blocks, each consumes L transmissions. Hence, the total relay transmissions in the network is $KL = \Theta(K \log K)$ by Theorem 1. Because there are $N = \Theta(K)$ relay nodes, each relay nodes needs to transmit $\Theta(\log K)$ times. ■

Lemma 1: In the original protocol, each relay node transmits at least $\Theta(K)$ data packets.

Proof: The destination needs to obtain at least K coded packets from K meetings with other nodes to decode all data. During such time period, each relay node behaves identically to the destination and meet at least K nodes on average. In the original protocol, each relay node transmits a coded packet whenever it meets another node. Hence, a relay node transmits at least K coded packets on average. ■

In Theorem 1, we assume each relay node has buffer size K to ensure that the spreading of different pseudo source packets are independent for ease of explanation. However, the buffer size can be significantly smaller. As shown in Corollary 1, each relay node transmits $\Theta(\log K)$ data packets. Therefore, it is easy to see the buffer size of each relay nodes needs to be only $\Theta(\log K)$ to ensure all transmission from it are linearly independent. Furthermore, the required relay buffer size can be reduced further because the coded packets received by a node arrive at different times. For example, suppose a node has buffer size 1. If all coded packets arrive at this node at the same time, it always generates linearly dependent coded packets. On the other hand, if two packets arrive at different times, *e.g.*, packet a arrives earlier than packet b, the node generates the coded packets linearly related to packet a before packet b arrives, whereas it produces a linear combination of packet a and b after packet b is received. The coded packets before and after packet b arrives is hence linearly independent. In the experiments of Sec. VI-C, we will show that the efficient protocol requires relay buffers with size only slightly larger than 1[2].

V. Priority Coding Protocol

In the network coding based protocol, the destination has to wait for a sufficient number of coded packets before decoding any useful data, despite its superiority over practical replication based protocols under limited network resource. In this section, we first introduce a simple priority coding protocol such that a subset of data, *i.e.*, the high priority data, can be decoded much earlier than the time to decode all data. We then use our analytical framework to study the trade-off in designing such a protocol.

We assume the K packets in the source can be classified into M different priority levels in descending levels of urgency — the packets in the ith level are more preferable and are decoded before the packets in the jth level, if $i < j$. The number of packets in the ith level is denoted by K_i, where $1 \leq i \leq M$. We further assume through layered coding [31] or particular application semantics, the number of packets K_i in each level can be adjusted to improve the utility of the application under our priority coding protocol. To make the analysis independent of application details, we assume the sum of the number of packets in all priority levels remains constant after adjusting K_i.

Next, we describe our priority protocol. *First*, the source transmits the data in the 1st level using the network coding based protocol as described in Sec. III-B. *Second*, after the destination decodes all data in the 1st level, the destination propagates an ACK towards the source by replicating the ACK whenever two nodes meet. *Third*, upon receiving the ACK, the source starts to transmit the data in the 2nd level with the same protocol as used in transmitting the data in the 1st level. Since the data in the 1st level has arrived at the destination, a node drops the data in the 1st level whenever the buffer is full and new data in the 2nd level arrive. *Finally*, such process

[2] For the interested reader, further discussions on efficient network coding protocols can be found in [30].

continues until the destination decodes the data in all priority levels.

We proceed to investigate the effectiveness and overhead of the above priority coding protocol by our analytical framework. It is easy to see in the priority protocol that the transmission process of the data in a priority level is identical to the network coding based protocol described in Sec. III-B. Therefore, we can use our analytical framework to compute the expected delivery delay of the data within any priority level. In particular, the delivery delay distribution of K_i packets, $F_{K_i}(t)$, can be computed with (5) by replacing K with K_i, and the expected delay $E[\tau_i]$ for the data in the ith level can be computed from $F_{K_i}(t)$. Next, we notice that the expected delay $E[\tau_{ACK}]$ in transmitting an ACK is equivalent to transmitting a packet, under the condition of infinite bandwidth, infinite buffer, and the replication based epidemic routing, which has been derived in [20]. Hence, we have

$$E[\tau_{ACK}] = \ln(N+1)/(\lambda N), \tag{23}$$

where N is the total number of relay nodes, and λ is the inter meeting rate of any pair of nodes in the network. Because the delivery process is composed of the data transmissions for M priority levels and the $M-1$ ACK transmissions interleaved among them, we can compute the total expected delay $E[\tau]$ to deliver all data as follows:

$$E[\tau] = \sum_{i=1}^{M} E[\tau_i] + (M-1)E[\tau_{ACK}], \tag{24}$$

where $E[\tau_{ACK}]$ is given in (23), and $E[\tau_i]$ is given in (5) by replacing K with K_i.

In Sec. VI-D, we use numerical analysis to show the effectiveness of the proposed priority coding scheme under a particular set of parameters. More importantly, we believe this simple scheme is effective in a wide range of parameters due to the following reason. Although the network has the same characteristic for both ACK and data packet transmissions, the amount of transmission opportunities required to deliver an ACK is much smaller than delivering all data packets in a priority level. Furthermore, the network is indeed the bottleneck, given the limited bandwidth when two nodes meet. Therefore, the delivery delay for an ACK is much shorter than the delivery delay of all data packets in a priority level. Hence, the priority coding scheme incurs only a modest amount of increased delivery delay due to the additional propagation delay of ACK packets and the first data packet (with the similar delay as ACKs) of each priority level as illustrated by the numerical analysis in Sec. VI-D.

VI. Model Validations and Performance Evaluation

In this section, we use simulations to verify the accuracy of our mathematical analysis. We show that our analytical results can demonstrate the advantage of the network coding based protocol over the replication based protocol, especially when bandwidth and buffer are limited. We further demonstrate the advantage of the proactive stopping mechanism in our efficient variant of the network coding based protocol. Finally, we show that the proposed priority coding based protocol is effective and induces low delay overhead.

We have developed a discrete-event simulator with the implementation of epidemic routing and network coding. To mitigate randomness in simulations, we show, for each data point in all figures, the average and the 95% confidence intervals from 100 independent experiments. We set the node meeting rate λ to 0.005 and the number of packets K to 10 in most experiments unless explicitly pointed out. In all simulations, we use GF(2^8) as the Galois field where network coding is operated.

A. Delivery Delay

1) Validation for Delivery Delay Distribution: We first validate the accuracy of our analysis on the distribution of packet delivery delay. We set the number of data packets $K = 10$, the number of relay nodes $N = 200$, the buffer size $B = 1$, and conduct 1000 simulations with independent random seeds for epidemic routing with both network coding and replication. In this set of experiments, we choose $B = 1$ because, as we will show in Sec. VI-A3, this is a reasonable value to use for epidemic routing with network coding. We observe similar results for other values of B, which are omitted to reduce redundancy. Fig. 2 plots the empirical and analytical CDF of delivery delay $F_K(t)$, derived from (5). For clarity in the figure, we show only the global rarest policy for the replication case since it is the ideal replication strategy and closest to the analysis. The local rarest and random policy are discussed later in Sec. VI-A2 and Sec. VI-A3. We notice that the analytical result is very close to the simulation result. Furthermore, the analytical delivery delay is slightly shorter than the empirical result since we have made several idealized assumptions in the analysis of Sec. III.

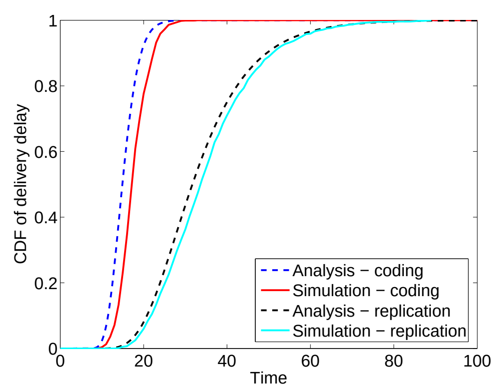


Fig. 2. CDF of delivery delay.

2) The Case for Limited Bandwidth: We then study the impact of the number of relay nodes on the delivery delay of K packets. The amount of bandwidth denotes the number of packets that can be exchanged between two nodes when they meet. It is easy to see that replication based epidemic routing achieves the minimal delivery delay when the bandwidth and buffer are sufficient to transmit K packets and hold K packets, respectively. Therefore, there is no advantage of network coding over replication when network resources are abundant.

In this paper, we focus on studying the difference between network coding and replication when the bandwidth b is only sufficient to transmit one data packet when two nodes meet as explained in Sec. II. The more general case when $1 < b \leq K$ can be easily extended and is outlined at the end of Sec. III-B.

We set $K = 10$, and the relay buffer size to 10 as well such that the buffer is sufficient to hold all K packets on each relay node. In Fig. 3, we plot the delivery delay as a function of the number of relay nodes. The analytical curve is the expected value computed from the CDF $F_K(t)$ of the delay distribution derived in (5) of Sec. III. The simulation curve is plotted from the average and confidence interval of 100 independent simulations as explained previously. We observe that the analytical results are close to the simulation results for both the network coding based protocol and the replication based protocols with the global rarest policy.

The delivery delay of the random policy is larger than the delivery delay of the global rarest policy since the assumption, that the packets on a node are uniformly distributed among all K packets, is less accurate in this case. The delivery delay of the local rarest policy is much larger than that of the random and global rarest policies. This shows that local counters do not provide an accurate estimation of the proportion of packets in the entire network. One may imagine if the nodes use the average of the local packet counters of the last several nodes it meets and its own counters, it could obtain a more accurate estimation. In the following, we omit the experimental results for the local rarest policy.

Fig. 3 also shows that the delivery delay of the network coding based protocol and the replication based protocol with the idealized global rarest policy are very close. This illustrates that network coding can achieve an even distribution of all packets as in the ideal replication based protocol. We emphasize that the practical replication based protocols, *i.e.*, with the random or local rarest policy, both have significantly longer delivery delay than the network coding based protocol.

Finally, Fig. 3 confirms our expectation that the delivery delay decreases as the number of relay nodes increases, because more relay nodes can expedite more transmissions from the source to the destination.

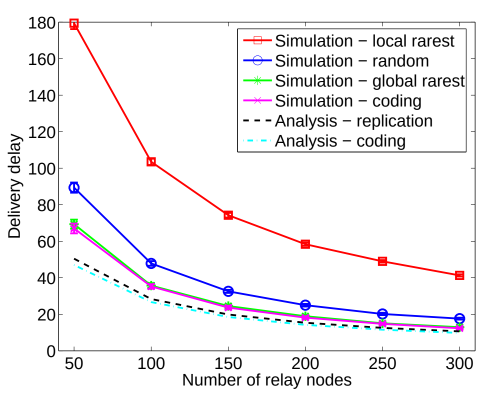


Fig. 3. Delivery delay under different numbers of relay nodes.

3) The Case for Both Limited Bandwidth and Buffer: We proceed to study the impact of the relay buffer size on the delivery delay. We set the number of relay nodes to 100 in this set of experiments and adjust the relay buffer size from 1 to 10. Fig. 4 shows that our analysis agrees with the simulation results for the network coding based protocol and the replication based protocol with the global rarest policy.

In addition, we note that both the analytical and simulation results demonstrate the benefit of network coding under limited buffer: the delivery delay of the network coding based protocol is not influenced by the buffer size, whereas the delivery delay of the replication based protocols increases significantly when the buffer size decreases. Such performance degradation of the replication based protocols is due to the coupon collector effect [32]. If we consider the extreme case that each buffer can store only one packet, assuming that the packet in a buffer is uniformly randomly chosen from the K packets, the coupon collector effect dictates that the destination node needs to collect $O(K \ln K)$ packets in order to obtain all K packets. On the other hand, under the same setting, the destination in the network coding based protocol can decode all K source packets from K coded packets with high probability.

Finally, we observe that the delivery delay of a practical replication based protocol, with the random policy, increases much more significantly than one with the global rarest policy when the buffer size decreases. This is because under the random policy, the packet distribution in node buffers deviates from the uniform distribution. If the node buffer size is K, such bias does not have as much impact after most nodes have collected all packets. However, if the buffer size is very small, such bias has a stronger influence throughout the delivery process and degrades the protocol performance.

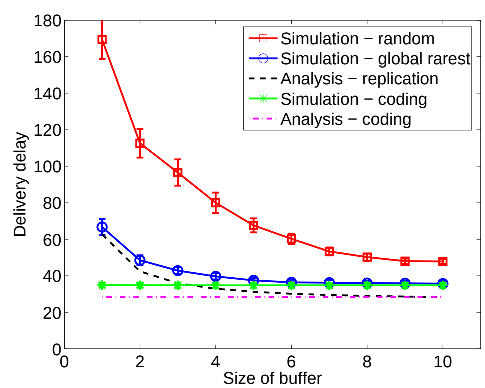


Fig. 4. Delivery delay under different buffer sizes.

B. Reactive Stopping and the Network Resource Usage

In this section, we study the network resource usage for both protocols under the reactive recovery schemes IMMUNE and VACCINE. We set the number of packets $K = 10$ and the number of relay nodes $N = 100$. We further set the relay buffer size B to 1 for the network coding based protocol and 8 for the replication based protocol since the performance of the replication based protocol will degrade if the relay buffer size is small. We trace the total number of buffered data packets in the entire network and the amount of transmissions by all relay nodes over the duration of simulation. To present clean figures, we show the simulation results of only the global rarest policy for the replication based protocol.

Fig. 5(a) and (c) demonstrate that the analytical results are very close to the simulation result for IMMUNE. For VACCINE, as shown in Fig. 5(b) and (d), the analysis is less accurate due to the well-known exponential amplification of modeling errors introduced by ACK flooding [20]. However, the analysis still captures the difference between the network coding based and replication based protocols.

In general, our analytical and simulation results show two observations. First, VACCINE is much more efficient than IMMUNE to reduce the amount of transmissions and to clean buffers. Second, the network coding based protocol requires much smaller buffers than the replication based protocol does. However, the amount of transmissions by the network coding based protocol is slightly larger than that of the replication based protocol, because at each node meeting, the network coding based protocol transmits a coded packet as long as there are coded packets in buffers, whereas the replication based protocol transmits a packet only when a node has a new packet that the other node does not possess. However, we emphasize that the amount of transmissions by the network coding based protocol can be drastically reduced by using the proactive stopping mechanism as shown in Sec. VI-C.

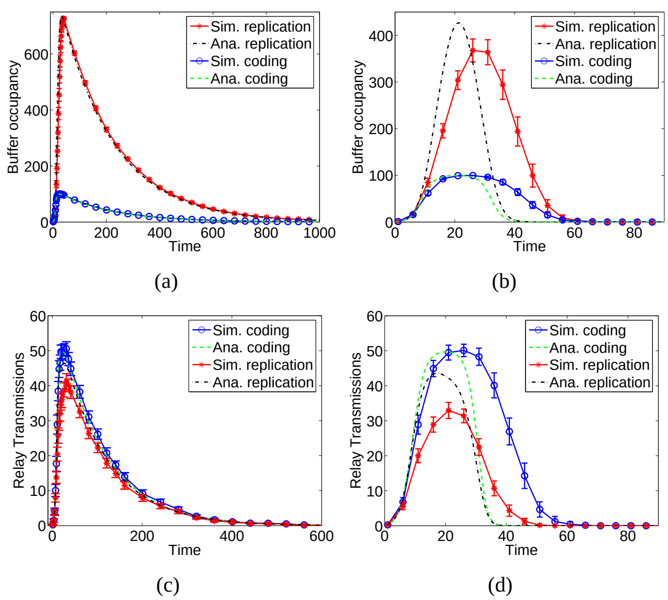


Fig. 5. (a) The buffer consumption of IMMUNE in time. (b) The buffer consumption of VACCINE in time. (c) The number of relay transmissions of IMMUNE in time. (d) The number of relay transmissions of VACCINE in time.

C. Proactive Stopping: Validation of Protocol Efficiency

In the following, we use simulation to illustrate the effectiveness of the more efficient network coding based protocol with proactive stopping. We set the number of source packets to 100, the number of relay nodes to 200, the number of pseudo source packets to 105, and the maximal relay buffer size to 100. We vary the maximal spray counter from 1 to 36 and compare the experimental results with the original protocol described in Sec. III-B under the same experiment settings. The results are shown in Fig. 6.

As expected, Fig. 6(a) shows that the amount of relay transmissions increases linearly as the spray counter increases, and that for the range of spray counter under consideration, the efficient protocol significantly reduces the amount of transmissions. More importantly, the efficient protocol can achieve near optimal performance. From Fig. 6(b), we observe that the packet delivery delay decreases significantly when the maximal spray counter increases. This is because the number of coded packets that the receiver needs to decode all data decreases dramatically until it is close to the number of coded packets required for decoding in the original protocol.

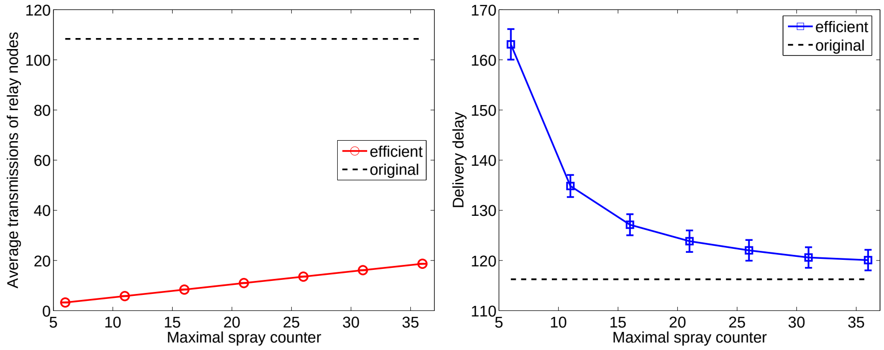
Next, we investigate the impact of the relay buffer size on the packet delivery delay. We set the maximal spray counter to 10 or 25 in two sets of experiments while varying the relay buffer sizes from 1 to 100. All the other settings are the same as the previous experiments. Fig. 7 shows that as long as the relay buffer size is more than 2, the performance of the efficient protocol is almost the same as the case with buffer size 100. This confirms our analysis in Sec. IV-B that the relay buffer sizes can be very small.

D. Priority Coding Advantage

In the following, we conduct numerical analysis on the performance of the priority protocol proposed in Sec. V. We study the simplest case, where only two priority levels exist. We set the total number of relay nodes N to 200. We further set the total number of packets to be transmitted to 100. We perform a set of numerical analysis by adjusting the number of packets in the high priority level from 1 to 99, and compare the delivery delay of the priority coding protocol with the original network coding based protocol, where all 100 packets are sent through the network in one priority level altogether.

Fig. 8 shows that our priority coding protocol is effective. It reduces the delivery delay of high priority packets while only slightly increasing the total delivery delay. For example, if the high priority level has 10 packets, the network delivers them with delay 14.9473, which is much smaller than the total data delivery delay, 104.3826, in the original protocol. Furthermore, the total delivery delay, 114.6457, in the priority coding protocol is only 10.26% larger than the data delivery delay in the original protocol. We further study the overhead of the priority coding protocol with more details in the following.

From Fig. 8, we observe that the delivery delay of high priority data is almost in linear relation with the number of packets with high priority. Such observation shows that the delivery delay in the network coding based protocol is composed of two types of components: the delivery delay of the first packet (5.1928, the first dot of the “high priority” curve in Fig. 8) and the delivery delay of the remaining packets, where the delivery delay of each packet is almost identical (about 0.9945) and much shorter than the delay of the first packet. This is because the transmission of the first packet incurs a delay with approximately the length of the shortest opportunistic path. Afterwards, the delivery delay of



(a) (b)

Fig. 6. (a) Average number of transmissions by a relay node vs. maximal spray counter. (b) Packet delivery delay vs. maximal spray counter.

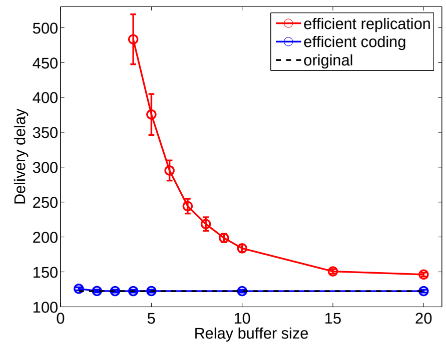


Fig. 7. Packet delivery delay under different sizes of relay buffers.

each packet is around the expected time $E[T_m]$ in which the destination meets another node because the destination can obtain an innovative coded packet from each contact with another node with high probability. We further confirm this by noting that $E[T_m] = \frac{1}{\lambda} \cdot \frac{1}{N+1} = 1/(0.005 * 201) = 0.995$, since $\frac{1}{\lambda}$ is the expected delay that two nodes meet. This is in agreement with the value observed in Fig. 8.

Because the delivery delay of each packet (excluding the first packet) is identical for both the priority protocol and the original network coding based protocol, it is easy to see that transmitting data in two priority levels separately will induce a delay overhead equaling the delivery delay of the first packet. Hence, the overhead of the priority coding protocol consists of two parts: the ACK propagation delay and the delivery time of the first packet. Note that in the analysis, the ACK propagation delay has similar delivery delay, 5.3033, as that of the first packet because both of them incur transmission delay as the length of the shortest opportunistic path approximately. Therefore, the delay overhead is low when there are two priority levels, because the ACK propagation delay and the delivery delay of the first packet are much smaller than the delivery delay of all packets. It can be expected that when we increase the number of priority levels, the overhead of the priority protocol increases. The quantitative relation of the protocol overhead and the number of priority levels can be easily estimated by our analytical framework. We omit the results of such obvious analysis due to space constraint.

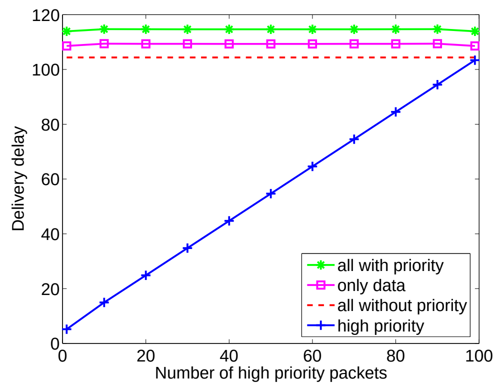


Fig. 8. Delivery delay under different numbers of packets in the high priority level. The plot labeled "only data" represents the sum of the delivery delay in two priority levels without the ACK packet.

VII. Conclusion

In this paper, we introduce a stochastic analysis framework to study the performance of epidemic routing with network coding in disruption tolerant networks. The analysis also includes the degenerate case of simple replication without network coding. Our analytical results quantify the superiority of network coding over simple replication under limited bandwidth and node buffer. Both reactive and proactive stopping mechanisms are considered, and their impact on the data delivery delay and network resource utilization are analyzed. Our analytical models are sufficiently accurate for examination of the tradeoffs involved in new protocol designs for opportunistic networks. Noting the slow-start disadvantage of network coding and inspired by our analysis of the data delivery delay, we propose a simple priority coding protocol, which can decode urgent data with much smaller delay than the baseline epidemic routing with network coding. Through our analytical model, we show that the priority coding protocol is effective and induces low delay overhead.

References

[1] Y. Lin, B. Liang, and B. Li, "Performance Modeling of Network Coding in Epidemic Routing," in *Proc. of the First ACM International Workshop on Mobile Opportunistic Networking (MobiOpp)*, 2007.

[2] A. Vahdat and D. Becker, "Epidemic Routing for Partially-Connected Ad Hoc Networks," Duke Univ., Tech. Rep. CS-200006, 2000.

[3] 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.

[4] 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.

[5] P. A. Chou, Y. Wu, and K. Jain, "Practical Network Coding," in *Proc. of 41th Annual Allerton Conference on Communication, Control and Computing*, October 2003.

[6] S. Jain, K. Fall, and R. Patra, "Routing in a Delay Tolerant Network," in *Proc. of ACM SIGCOMM*, 2004.

[7] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks," in *Prof. of IEEE INFOCOM*, 2006.

[8] W. Zhao, M. Ammar, and E. Zegura, "A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks," in *Proc. of the Fifth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2004.

[9] Y. Wang, S. Jain, M. M. tonosi, and K. Fall, "Erasure-Coding Based Routing for Opportunistic Networks," in *Proc. of ACM SIGCOMM Workshop on Delay Tolerant Networking and Related Topics (WDTN)*, Philadelphia, PA, USA, 2005.

[10] S. Jain, M. Demmer, R. Patra, and K. Fall, "Using Redundancy to Cope with Failures in a Delay Tolerant Network," in *Proc. of ACM SIGCOMM*, Philadelphia, Pennsylvania, USA, 2005.

[11] L.-J. Chen, C.-H. Yu, T. Sun, Y.-C. Chen, and H. hua Chu, "A Hybrid Routing Approach for Opportunistic Networks," in *Proc. of ACM SIGCOMM Workshop on Challenged Networks*, 2006.

[12] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in The Air: Practical Wireless Network Coding," in *Proc. of ACM SIGCOMM*, 2006.

[13] C. Fragouli, J. Widmer, and J.-Y. L. Boudec, "A Network Coding Approach to Energy Efficient Broadcasting: from Theory to Practice," in *Proc. of IEEE INFOCOM*, 2006.

[14] 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.

[15] C. Fragouli, J. Widmer, and J.-Y. L. Boudec, "On the Benefits of Network Coding for Wireless Applications," in *NetCod*, 2006.

[16] J. Widmer and J.-Y. L. Boudec, "Network Coding for Efficient Communication in Extreme Networks," in *Proc. of ACM SIGCOMM Workshop on Delay Tolerant Networking and Related Topics (WDTN)*, Philadelphia, PA, USA, 2005.

[17] Z. J. Haas and T. Small, "A New Networking Model for Biological Applications of Ad Hoc Sensor Networks," *IEEE/ACM Transactions on Networking*, vol. 14, no. 1, pp. 27–40, February 2006.

[18] T. Small and Z. J. Haas, "Quality of Service and Capacity in Constrained Intermittent-Connectivity Networks," *IEEE Transactions on Mobile Computing*, vol. 6, no. 7, 2007.

[19] R. Groenevelt, P. Nain, and G. Koole, "Message Delay in Mobile Ad Hoc Networks," in *Proc. of PERFORMANCE*, October 2005.

[20] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, "Performance Modeling of Epidemic Routing," *Elsevier Computer Networks*, 2007.

[21] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Efficient Routing in Intermittently Connected Mobile Networks: The Multi-copy Case," *IEEE/ACM Transaction on Networking*, 2007.

[22] M. Mitzenmacher, "The Power of Two Choices in Randomized Load Balancing," *IEEE Transactions on Parallel and Distributed Computing*, vol. 12, no. 10, pp. 1094–1104, 2001.

[23] D. Qiu and R. Srikant, "Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks," in *Proc. of SIGCOMM*, Portland, Oregon, 2004.

[24] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, "On the Benefits of Random Linear Coding for Unicast Applications in Disruption Tolerant Networks," in *Second Workshop on Network Coding, Theory, and Applications (NetCod)*, 2006.

[25] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of Human Mobility on the Design of Opportunistic Forwarding Algorithms," in *Proc. of IEEE INFOCOM*, Barcelona, Spain, 2006.

[26] T. Karagiannis, J.-Y. L. Boudec, and M. Vojnovic, "Power Law and Exponential Decay of Inter Contact Times Between Mobile Devices," in *Proc. of ACM MOBICOM*, 2007.

[27] H. Cai and D. Y. Eun, "Crossing Over the Bounded Domain: From Exponential To Power-law Inter-meeting Time in MANET," in *Proc. of ACM MOBICOM*, 2007.

[28] B. Fan, D.-M. Chiu, and J. C. Lui, "Stochastic Analysis and File Availability Enhancement for BT-like File Sharing Systems," in *Fourteenth IEEE International Workshop on Quality of Service (IWQoS)*, Yale University, New Haven, CT, USA, 2006.

[29] A. G. Dimakis, V. Prabhakaran, and K. Ramchandran, "Decentralized Erasure Codes for Distributed Networked Storage," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2809–2816, June 2006.

[30] Y. Lin, B. Li, and B. Liang, "Efficient Network Coded Data Transmissions in Disruption Tolerant Networks," in *Proc. of IEEE INFOCOM*, 2008.

[31] K. Sayood, *Introduction to Data Compression*, 3rd ed. Morgan Kaufmann, 2006.

[32] M. Mitzenmacher and E. Upfal, *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge U., 2005.

APPENDIX A
DERIVATION OF ODES (5) TO COMPUTE THE DELAY DISTRIBUTIONS

It is easy to see the event $\{T_i > t + \delta t\}$ implies the event $\{T_i > t\}$. Hence, we have $\{T_i > t + \delta t, T_i > t\} = \{T_i > t + \delta t\}$ and

$$\begin{aligned}\Pr(T_i > t + \delta t) &= \Pr(T_i > t + \delta t, T_i > t)\\ &= \Pr(T_i > t)\Pr(T_i > t + \delta t | T_i > t)\\ &= \Pr(T_i > t)(1 - \Pr(T_i < t + \delta t | T_i > t)) \ .\end{aligned} \tag{25}$$

Next, we derive $\Pr(T_i < t + \delta t | T_i > t)$ in (25). In a short time interval δt, the probability that more than one packet arrives can be ignored compared with the probability that one packet arrives. Hence, the event $\{T_i < t + \delta t\}$ that the destination receives i packets before time $t + \delta t$ happens only if the destination receives $i-1$ packets before t, *i.e.*, the event $\{T_{i-1} < t\}$. Therefore, we have

$$\Pr(T_i < t + \delta t | T_i > t) \simeq \delta t D_{i-1} \Pr(T_{i-1} < t | T_i > t), \tag{26}$$

where D_{i-1} ((3)) is the receiving rate of the destination when it has $i-1$ packets, *i.e.*, the probability that the destination obtains an innovative packet in a small interval δt.

We then derive $\Pr(T_{i-1} < t | T_i > t)$ as follows:

$$\begin{aligned}\Pr(T_{i-1} < t | T_i > t) &= 1 - \Pr(T_{i-1} > t | T_i > t)\\ &= 1 - \frac{\Pr(T_{i-1} > t, T_i > t)}{\Pr(T_i > t)}\\ &= 1 - \frac{\Pr(T_{i-1} > t)}{\Pr(T_i > t)},\end{aligned} \tag{27}$$

where the third equality holds since $T_i \geq T_{i-1}$, the time to receive i coded packets is always greater than or equal to the time to receive $i-1$ coded packets. Substituting (27) into (26), and (26) into (25), we get

$$\Pr(T_i > t + \delta t) = \Pr(T_i > t)(1 - \delta t D_{i-1}(t)(1 - \frac{\Pr(T_{i-1} > t)}{\Pr(T_i > t)})) \ . \tag{28}$$

Therefore, we can compute the derivative of $F_i(t)$ by

$$\begin{aligned}\frac{dF_i}{dt} &= \lim_{\delta t \to 0} \frac{F(t + \delta t) - F(t)}{\delta t}\\ &= \lim_{\delta t \to 0} -\frac{\Pr(T_i > t + \delta t) - \Pr(T_i > t)}{\delta t}\\ &= \lim_{\delta t \to 0} \Pr(T_i > t) \frac{\delta t D_{i-1}(t)(1 - \frac{\Pr(T_{i-1} > t)}{\Pr(T_i > t)})}{\delta t}\\ &= D_{i-1}(t)(\Pr(T_i > t) - \Pr(T_{i-1} > t))\\ &= D_{i-1}(t)(F_{i-1}(t) - F_i(t)),\end{aligned} \tag{29}$$

where the third equality holds by substituting with (28). Similarly, we can derive $\frac{dF_1}{dt} = D_0(t)(1 - F_1(t))$. Hence, the CDF $F_K(t)$ can be computed by solving the ODEs in (5).

Appendix B
Derivation of Relay Transmissions for the Replication Based Protocol

In this appendix, we analyze the amount of relay transmissions in the network with the state of data packets $X_i(t)$ and ACKs $Z_i(t)$, derived in Sec. III-C for the replication based protocol. We examine the details when two relay nodes meet, *e.g.*, node a with i data packets and m ACKs meets node b with j data packets and n ACKs. The number of data packets refers to the number of original data packets including those removed by ACKs. To derive the expected number of relay data transmissions, we first compute the probability that node a transmits a data packet to node b.

First, we investigate the case where $m \leq n$. As we discussed previously in Sec. IV-A2, we have $A_a \subseteq A_b$, where A_a and A_b refer to the ACK sets on node a and b, respectively. Hence, we first compute the expected remaining data packets in both nodes that are removed by ACKs in A_a. Let D_a and D_b denote the remaining data packets in node a and b, respectively. By the discussion in Sec. IV-A2, we have the expected size of D_a and D_b, denoted as d_a and d_b, respectively:

$$\begin{aligned} d_a &= i(1 - \frac{m}{K}), \\ d_b &= j(1 - \frac{m}{K}). \end{aligned} \tag{30}$$

We assume the remaining packets in D_a, D_b, and $A_b - A_a$ are uniformly distributed among the $K - m$ remaining data packets (after removing the packets in A_a), denoted as R. If D_a contains a new packet not in $D_b \cup (A_b - A_a)$, *i.e.*, $D_a - (D_b \cup (A_b - A_a)) \neq \Phi$, node a will transmit a data packet to node b. Based on this, we derive the probability that a data transmission occurs at node a.

Before computing the probability of the event $D_a - (D_b \cup (A_b - A_a)) \neq \Phi$, we first compute the expected number of packets in $D_b \cup (A_b - A_a)$. As such computation will be used multiple times, we calculate the general case: the expected number of packets in $X \cup Y$, where the packets in X and Y are independently uniformly distributed among Z, given $X \subseteq Z$ and $Y \subseteq Z$. Without loss of generality, we assume that $|X| \geq |Y|$. Such computation can be further divided into two cases. We first consider the case where $|X| + |Y| \leq |Z|$. For every packet p, where $p \in Y$, it has probability $(|Z| - |X|)/|Z|$ not to be in X, because the uniform distribution assumption. Hence on expectation, there are $|Y|(|Z| - |X|)/|Z|$ packets in $Y - X$. Therefore, the expected number of packets in $X \cup Y$ is $|X| + \frac{|Y|(|Z|-|X|)}{|Z|}$. Similarly, we can compute the case when $|X| + |Y| > |Z|$. In summary, we have the following function $f(|X|, |Y|, |Z|)$ to obtain the expected number of packets in the set $X \cup Y$:

$$f(|X|,|Y|,|Z|) = \begin{cases} |X| + \frac{|Y|(|Z|-|X|)}{|Z|} & \text{if } |X| + |Y| \leq |Z|, \\ |X| + \frac{(|Z|-|X|)^2}{2|Z|-|X|-|Y|} & \text{if } |X| + |Y| > |Z|. \end{cases} \tag{31}$$

Hence, the expected number of packets in the set $D_b \cup (A_b - A_a)$ is

$$U_b = f(d_b, n - m, K - m) \tag{32}$$

where d_b is given in (30), and $f(\cdot,\cdot,\cdot)$ is defined in (31). Therefore, the probability that node a transmits a data packet to node b is the probability $g(i, m, j, n)$ that node b can obtain a new packet from node a:

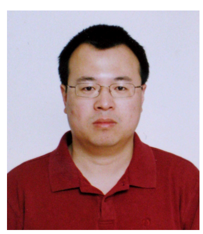
$$g(i, m, j, n) = \Pr(U_b, d_a) \tag{33}$$

where the function $\Pr(\cdot,\cdot)$ is defined in (7), and U_b is derived in (32).

Because the data packets and ACKs are assumed independent on a node as we discuss in Sec IV-A2, a node with i packets has probability $Z_m(t)/N$ to have m ACKs. Similarly, a node with j data packets has probability $Z_n(t)/N$ to have n ACKs. Furthermore, the meeting rate of two nodes is λ. Therefore, we can compute the expected number $D(t)$ of relay transmissions by summing all cases as follows:

$$D(t) = \lambda \sum_{i=0}^{B} \sum_{m=0}^{K} \sum_{j=0}^{B} \sum_{n=0}^{K} X_i(t) \frac{Z_m(t)}{N} X_j(t) \frac{Z_n(t)}{N} g(i, m, j, n). \tag{34}$$

For the second case where $m > n$, similarly, we first remove the data packets by ACKs in the smaller ACK set A_b. Denoting the remaining data packets on node a and b with D'_a and D'_b, respectively, we have the number of data packets d'_a and d'_b in them similarly as derived in (30). The probability that node a will transmit a data packet to node b is the probability of the event that D'_a has a new packet which is not in the set $D'_b \cup (A_a - A_b)$. Therefore, the expected number of transmissions can be obtained similarly as in the first case where $m \leq n$. We omit the details of the analysis here due to space constraint.



Yunfeng Lin. Yunfeng Lin received his B.Engr. degree from Department of Computer Science and Technology, Tsinghua University, China, in 2000, and his M.Math. degree from School of Computer Science, University of Waterloo, Canada, in 2005. He is currently a Ph.D. candidate at the Department of Electrical and Computer Engineering, University of Toronto, Canada. His current research interest includes performance modeling and distributed algorithm design in wireless networks.

Baochun Li. Baochun Li received his B.Engr. degree in 1995 from Department of Computer Science and Technology, Tsinghua University, China, and his M.S. and Ph.D. degrees in 1997 and 2000 from the Department of Computer Science, University of Illinois at Urbana-Champaign. Since 2000, he has been with the Department of Electrical and Computer Engineering at the University of Toronto, where he is currently an Associate Professor. He holds the Nortel Networks Junior Chair in Network Architecture and Services since October 2003, and the Bell University Laboratories Chair in Computer Engineering since July 2005. In 2000, he was the recipient of the IEEE Communications Society Leonard G. Abraham Award in the Field of Communications Systems. His research interests include application-level Quality of Service provisioning, wireless and overlay networks. He is a senior member of IEEE, and a member of ACM.

Ben Liang. Ben Liang received honors simultaneous B.Sc. (valedictorian) and M.Sc. degrees in electrical engineering from Polytechnic University in Brooklyn, New York, in 1997 and the Ph.D. degree in electrical engineering with computer science minor from Cornell University in Ithaca, New York, in 2001. In the 2001 - 2002 academic year, he was a visiting lecturer and post-doctoral research associate at Cornell University. He joined the Department of Electrical and Computer Engineering at the University of Toronto in 2002, where he is now an Associate Professor. His current research interests are in mobile networking and multimedia systems. He won an Intel Foundation Graduate Fellowship in 2000 toward the completion of his Ph.D. dissertation and an Early Researcher Award (ERA) given by the Ontario Ministry of Research and Innovation in 2007. He was a co-author of the Best Paper Award at the IFIP Networking conference in 2005 and the Runner-up Best Paper Award at the International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks in 2006. He is a senior member of IEEE and a member of ACM and Tau Beta Pi. He is an associate editor for the Wiley Security and Communication Networks journal and serves on the organizational and technical committees of a number of conferences each year.