Performance Analysis of Network Coding with IEEE 802.11 DCF in Multi-Hop Wireless Networks

Somayeh Kafaie, *Student Member, IEEE*, Mohamed Hossam Ahmed, *Senior Member, IEEE*,
Yuanzhu Chen, *Member, IEEE*, and Octavia A. Dobre, *Senior Member, IEEE*

Abstract—Network coding is an effective idea to boost the capacity of wireless networks, and a variety of studies have explored its advantages in different scenarios. However, there is not much analytical study on throughput and end-to-end delay of network coding in multi-hop wireless networks considering the specifications of IEEE 802.11 Distributed Coordination Function. In this paper, we utilize queuing theory to propose an analytical framework for bidirectional unicast flows in multi-hop wireless mesh networks. We study the throughput and end-to-end delay of inter-flow network coding under the IEEE 802.11 standard with CSMA/CA random access and exponential back-off time considering clock freezing and virtual carrier sensing, and formulate several parameters such as the probability of successful transmission in terms of bit error rate and collision probability, waiting time of packets at nodes, and retransmission mechanism. Our model uses a multi-class queuing network with stable queues, where coded packets have a non-preemptive higher priority over native packets, and forwarding of native packets is not delayed if no coding opportunities are available. Finally, we use computer simulations to verify the accuracy of our analytical model.

Index Terms—Performance analysis, network coding, queuing networks, IEEE 802.11 DCF, multi-hop wireless networks

✦

1   INTRODUCTION

CAPACITY is a crucial resource in multi-hop wireless networks as it is shared not only between the source and destination of data packets but also among relay nodes forwarding packets. To increase the transmission capacity of wirless networks, the powerful concept of network coding [1] has been introduced, which can improve performance significantly in theory, without considering PHY/MAC layer constraints such as contention, collision and interference. However, network protocols inevitably deal with such physical phenomena and constraints. Therefore, more theoretical studies are needed to better quantify the benefits of network coding over traditional forwarding for actual protocols considering PHY/MAC layer specifications.

There have been many experimental studies on this subject, but much fewer mathematical analyses. Some previous theoretical studies are designed for saturated queues, where each node always has a packet to transmit that would cause an infinite delay. In many cases, researchers consider a simple topology, where source is one [2], [3], [4], [5] or two [6], [7], [8], [9], [10], [11], [12], [13], [14], [15] hops away from the

destination. Furthermore, the theoretical research on multi-hop networks [16], [17], [18], [19], [20], [21] usually models network coding with simplifying assumptions, such as conflict-free scheduled access, no interference, no collision, or no exponential back-off. Moreover, most research in this subject investigates only the throughput of the network, and postpone the transmission of native packets in favor of providing more coding opportunities.

The main contributions of this paper are as follows:

1) We apply multi-class queuing network to study the performance of multi-hop wireless mesh networks applying inter-flow network coding [22], where intermediate nodes can mix packets of different flows by bitwise *XOR* operation. This model provides an analytical framework for a multi-hop chain topology with bidirectional unicast flows in opposite directions. In contrast to other studies, no artificial delay is injected in forwarding native packets even if there is no coding opportunity. In fact, we do not postpone transmission of native packets artificially to generate coded packets (i.e., opportunistic coding). Also, we consider separate classes of queues for native and coded packets, while the coded queue is a higher-priority queue.

2) We develop our analytical framework for both non-coding and coding schemes in multi-hop wireless networks, and formulate not only the throughput but also the end-to-end delay in a stable network.

3) The proposed model takes into account PHY/MAC layer specifications. It applies random medium access CSMA/CA as in IEEE 802.11 Distributed Coordination Function (DCF) with binary exponential back-off

---
- *S. Kafaie, M.H. Ahmed, and O.A. Dobre are with the Department of Electrical and Computer Engineering, Faculty of Engineering and Applied Science, Memorial University, St. John's, NL A1B 3X5, Canada. E-mail: {somayeh.kafaie, mhahmed, odobre}@mun.ca.*
- *Y. Chen is with the Department of Computer Science, Memorial University, St. John's, NL A1B 3X5, Canada. E-mail: yzchen@mun.ca.*

considering clock freezing and virtual carrier sensing as explained in Sections 3.2 and 4.1.2. We consider retransmission, collision probability, link qualities and coding probabilities in calculating the throughput and upper-bound of average end-to-end delay of the network. Also, the validity of the analytical model is shown by simulations in NS-2.

The rest of this paper is organized as follows. Related work is discussed in Section 2. We explain the system model and assumptions in Section 3. Section 4 introduces our derived formulation of the throughput and end-to-end delay for the non-coding and coding schemes. To show the accuracy of our analytical model, we compare the results with computer simulations in Section 5. Finally, Section 6 draws conclusions, and discusses directions for future work.

2 BACKGROUND AND RELATED WORK

Prior theoretical studies on network coding usually consider a simple topology. Most of them study the performance for a two-way relay [6], [7], [8], [14], or derive some analytical bounds for a single relay in a two-hop region, where multiple sources initiate unicast sessions to multiple destinations [9], [10], [11], [12], [13], [15]. In particular, Amerimehr and Ashtiani [7] study the throughput and delay of a two-way relay by adopting frequency division duplexing (FDD). Without focusing on PHY/MAC layer constraints, they compare the throughput and delay in the relay for two cases where 1) the relay postpones transmission of native packets, and 2) native packets are sent immediately.

Sagduyu et al. study the stable throughput when one or two sources broadcast their packets to two destinations [2] or more [3] via independent channels. Paschos et al. [6] study a two-way relay in inter-flow network coding taking into account overhearing, where coding decisions at the relay are either stochastic or deterministic via receiving overhearing reports. Moghadam and Li [4], [5] study the maximum stable throughput in single-hop wireless networks, where a source multicasts data packets to several destinations directly, and network coding is applied to retransmit the packets not received by a subset of the destinations.

In addition, Jamali et al. propose a dynamic scheduling based on a threshold on the amount of information at nodes' transmission buffers in bidirectional relay networks. This scheduling is used to maximize throughput both without any constraint on the delay [24], and with constraint to guarantee a certain average delay [8]. Furthermore, Umehara et al. [14] analyze the throughput and delay of network coding in two-hop networks with two unbalanced traffic cases (i.e., one-to-one and one-to-many bidirectional relay) employing slotted ALOHA. They also extend the model to single-relay multi-user wireless networks and provide the achievable region in throughput [15].

In another single-relay research, Lin et al. [11] study the throughput of network-layer and physical-layer network coding under IEEE 802.11 DCF with two groups of nodes communicating with each other via a relay node. In a similar work, where again all nodes are in carrier sensing range of each other, they not only study the throughput under slotted ALOHA but also propose a hybrid network coding scheme (i.e., a combination of physical-layer and network-layer network coding) to improve performance [12].

Regarding multi-hop wireless networks, Sagduyu et al. [16] consider a collision-free scheduled access to formulate throughput for both saturated and non-saturated queues. However, in case of a random access scheme, their analytical model is limited to saturated queues. In this paper, instead of limiting nodes to scheduled access, we study the performance using IEEE 802.11 MAC layer, where collision can occur without assuming saturated queues. In addition, we provide simulation results to verify our model.

In a similar theoretical-based approach for multicast sessions, Amerimehr et al. [18] derive throughput for multi-hop wireless networks. They also define a new metric, *network unbalance ratio*, which identifies the amount of unbalance in stability among nodes. However, their estimate of service time does not take into account some important features of IEEE 802.11 DCF like binary exponential random back-off. Furthermore, they postpone transmission of the native packet at a node until receiving a packet from another flow to be combined with it, and thus, causing a long delay.

In another work considering IEEE 802.11 DCF, Lin and Fu [19] investigate the throughput capacity of physical-layer network coding in which a common center node exchanges packets with others in multi-hop wireless networks. They analyze such canonical networks both with equal and variable link-length, and find the optimal number of hops to maximize the throughput. In addition, Ko and Kim [23] study the throughput and end-to-end delay of multi-hop wireless networks utilizing IEEE 802.11 DCF only for traditional forwarding, when every node initiates a flow with the same rate to a random destination, and same arrival rate is assumed at all nodes. They derive a delay-constrained capacity in terms of carrier sensing range and packet generation rate.

Furthermore, Hwang et al. [20] propose an analytical framework for bidirectional unicast flows in multi-hop wireless networks. Their work considers collision and different interference levels in CSMA/CA by varying the carrier-sensing range and signal-to-interference ratio to maximize the throughput in different retransmission schemes. In comparison, the novelty of our work consists of the following: 1) our model formulates not only throughput but also end-to-end delay; 2) in our model, opportunistic coding is applied (i.e., if a node has a transmission opportunity, it does not delay forwarding native packets to generate coded packets); 3) our focus is on more realistic case of stable queues.

Table 1 presents an overview of all studies discussed in this section.

3 SYSTEM OVERVIEW

Before further explanation of the model, let us summarize the symbols used in this paper in Table 2.

3.1 Network Model and Assumptions

We propose an analytical model of network coding for bidirectional unicast flows in multi-hop wireless mesh networks to study the throughput and end-to-end delay. Our analytical results are provided for a chain topology with k nodes as depicted in Fig. 1, with two flows in opposite directions. As shown in this figure, N_1 and N_k transmit their packets to each other via intermediate nodes N_2 to N_{k-1}, while we

TABLE 1
An Overview of the Analytical Research in the Literature

Research	Network coding	Number of hops	Throughput	Delay	Stable queues	Random access	Unicast/ multidcast	Opportunistic coding	Exponential back-off
[2], [3]	√	1	√	-	√	√	multicast	-	-
[4], [5]	√	1	√	-	√	√	multicast	√	-
[6]	√	2	√	-	√	-	unicast	-	-
[7]	√	2	√	relay	√	-	unicast	both	-
[8]	√	2	√	√	√	√	unicast	-	-
[9], [12], [13]	√	2	√	-	√	√	unicast	-	-
[10]	√	2	√	-	√	priority/ equal access	unicast	both	-
[11]	√	2	√	-	√	√	unicast	-	√
[14], [15]	√	2	√	√	√	√	unicast	√	-
[16]	√	≥ 3	√	-	-	√	multicast	-	-
[18]	√	≥ 3	√	-	√	√	multicast	-	-
[19]	√	≥ 3	√	-	√	√	unicast	-	-
[20]	√	≥ 3	√	-	-	√	unicast	-	√
[23]	-	≥ 3	√	√	√	√	multicast	not applicable	√

assume that only N_{i-1} and N_{i+1} are in the transmission range of N_i.

In this model, we assume that nodes usually do not move, and packets of UDP flows from a source traverse multiple wireless hops to be received by the destination. Furthermore, we assume that a node cannot transmit and receive at the same time and the feedback channel is reliable; thus if a node does not hear an acknowledgement (ACK) on time, it assumes that the data packet is lost.

In this network, we consider each node as a queuing system, where the packets in the sending buffer are customers of the queue. We assume that the queues have an unlimited capacity and are in a stable state, i.e., the arrival rate is less than the service rate. When a node finds the channel idle, it sends a packet from the head of its queue. Hence, each node provides services with one server to the packets in its queue, and *Queuing Theory* can be used to model this network.

Our described network has all the properties of *open Jackson networks* [25], [26] including 1) each node is considered a queuing system; 2) the packet generation rate at source N_i $(i = 1, k)$ follows the Poisson model with a mean rate γ_i; 3) service time at node N_i $(i = 1, \ldots, k)$ is assumed independent from that of other nodes, and it is exponentially distributed with parameter μ_i; and 4) a packet that has completed service at node N_i (i.e., the packet has been transmitted) will go next to node N_j with probability $r_{i,j}$. This probability, presented in (1), for the next-hop equals successful transmission probability $p_{i,j}$, and for other nodes equals zero

$$r_{i,j} = \begin{cases} p_{i,j} & \text{if } N_j \text{ is a neighbor of } N_i \\ 0 & \text{elsewhere.} \end{cases} \quad (1)$$

To formulate this network, we employ concepts from the probability theory, queuing theory and Jackson networks [27], [28], [29]. Based on the *Burke's* Theorem [30], in a stable stationary queuing system, the departure process of an exponential server is Poisson if the arrival rate follows a Poisson process. Furthermore, the *Jackson's Theorem* states that in the Jackson network each node behaves as if its input were Poisson. Therefore, the arrival rate at other nodes, in addition to the sources, can be considered a Poisson process. We assume that λ_i denotes the arrival rate at N_i.

3.2 Data Link Layer Description

In this paper, the same data link layer signaling as IEEE 802.11 DCF [31] is applied, with CSMA/CA random access. At the beginning of each time slot, a node, with a packet to transmit, senses the channel. If the node finds the channel idle for a Distributed Inter-Frame Space (DIFS) period of time, it waits for a random back-off interval to minimize the probability of collision with packets transmitted by other nodes, and then transmits the packet. We consider a fixed number of time slots for the transmission time of each packet.

The random back-off interval in DCF is discrete with binary exponential growth. To transmit a new packet, random back-off is uniformly chosen from $[0, \text{CW}_{\min} - 1]$, where $\text{CW}_{\min}$ is the minimum contention window. When a packet is retransmitted for the mth time ($m > 0$), the contention window range will be extended to $[0, 2^m \text{CW}_{\min} - 1]$, while $2^m \text{CW}_{\min}$ is upper-bounded by $\text{CW}_{\max}$.

Based on the specifications of the IEEE 802.11 standard, the back-off and the DIFS counters are decremented as long as the channel is sensed idle. As soon as it is sensed busy, the node freezes the state of the clock and stops counting down until sensing the idle channel again. Therefore, although the value of DIFS and selected back-off (i.e., the number of ticks in the counter) are specified, the counter may pause due to another transmission which makes the channel busy. This "clock freezing" behaviour needs to be taken into account in calculating the back-off time.

The default feedback mechanism in the DCF is automatic repeat request (ARQ), where an ACK is transmitted by the receiver of the data packet, after a period of time called short inter-frame space (SIFS). Since the SIFS is shorter than the DIFS, no other node will sense the channel idle for a DIFS before the end of the ACK transmission. If the sender of a data packet does not receive an ACK before time-out, it will increase the back-off interval and retransmit the packet.

3.3 The Probability of Successful Transmission

We calculate the probability of successful transmission in each link in terms of the bit error rate (p_e) and collision. In general, a packet transmission, at the link between N_i and

TABLE 2
The Description of Used Symbols

Symbol	Description
$p_{i,j}$	probability of successful transmission from N_i to N_j
δ	maximum propagation delay
γ_i	packet generation rate at the source N_i
λ_i	arrival rate at N_i
μ	service rate of the queue
L_p	packet length
$\mathrm{CW}_{\min}$	minimum contention window
β	maximum number of transmissions of a packet at each node
T_{data}	transmission time of a packet
T_{ack}	transmission time of an acknowledgement
θ	throughput
DIFS	Distributed Inter-Frame Space
SIFS	Short Inter-Frame Space
T_{trans}	$T_{\mathrm{data}} + T_{\mathrm{ack}} + \mathrm{SIFS}$
$T_{\mathrm{counter}}(m)$	$\mathrm{DIFS} + \frac{2^{m-1}\mathrm{CW}_{\min}-1}{2}$
$T(m)$	time spent on DIFS and back-off in mth transmission by taking into account the "clock freezing" behavior
W	upper-bound of the average end-to-end delay
$W^{(j)}$	upper-bound of the average end-to-end delay of the jth flow
$T_s(m)$	service time at the mth transmission of a packet
$\lambda_{in,i}^{n(j)}$	arrival rate of native packets of the jth flow at N_i
$\lambda_{out,i}^{n(j)}$	output rate of native packets of the jth flow at N_i
$\lambda_{in,i}^{c(j)}$	arrival rate of coded packets of the jth flow at N_i
$\lambda_{out,i}^{c}$	output rate of coded packets at N_i
$\lambda_i^{n(j)}$	arrival rate of the jth flow in Q^n of N_i
λ_i^{c}	arrival rate in the coded queue of N_i
$W(Q)$	average waiting time in queue Q
W_{system}	average waiting time in the queuing system
$\bar{R}_i$	mean residual service time in priority queues at N_i
$\mu_i^{n,\mathrm{seen}}$	service time seen by lower priority queue, Q^n
$P_{\mathrm{mtc}}(r)$	probability that a packet from flow r in Q^n moves to Q^c
$N(r,w)$	average number of the packets of flow r arrived in Q^n during w time window
$N(r)$	average number of the packets of flow r ahead of the currently arrived packet in Q^n
$\pi_0(Q^{n(r)})$	probability of having no packets from flow r in Q^n
I_i	set of all nodes in interference range of N_i including N_i
h_x	probability that node x transmits a packet during transmission between two other nodes
$P_{i,j}^d$	probability that N_i drops a packet with next-hop N_j after failure in β transmissions
$P_{i,j}^{\mathrm{decode}}$	decoding probability of a coded packet arrived at N_j from N_i

N_j, may fail due to packet error rate or collision ($C_{i,j}$), where error rate of a packet with length L_p is approximated as $p_e \times L_p$. Thus, the probability of successful transmission of a packet from N_i to N_j can be calculated as

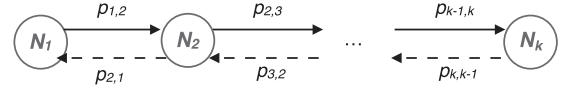$$p_{i,j} = (1 - C_{i,j})(1 - p_e \times L_p). \tag{2}$$

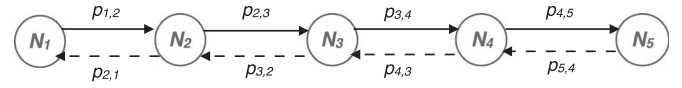Fig. 1. Chain topology used for the analytical model.

Fig. 2. Chain topology with five nodes.

We assume that the probability of collision between a data packet and an ACK is negligible; this is a valid assumption because: 1) the length of ACKs is significantly shorter than the length of data packets, and 2) ACKs are given higher priority and are sent earlier than any data packet. A transmission from N_i to N_j will fail if at the same time, N_j or any other node in its interference range transmits a packet. Let us denote I_j as the set of all nodes in the interference range of N_j, including N_j itself. Then the probability of successful transmission from N_i to its neighbor, N_j, can be computed as

$$p_{i,j} = (1 - p_e L_p) \prod_{N_x \in I_j - \{N_i\}} (1 - h_x), \tag{3}$$

where h_x represents the probability that N_x transmits a packet during packet transmission between two other nodes.

If N_i transmits a packet at time t, any node in its interference range will sense that the channel is busy after the propagation delay (δ), and avoid any transmission. Therefore, during a propagation delay window before and after N_i's transmission (i.e., $(t - \delta, t + \delta)$), other nodes may transmit their packet which will collide with N_i's transmission. Although the propagation delay depends on the distance, we assume a fixed propagation delay as the maximum propagation delay. The probability that N_x transmits a packet during this time window can be estimated as $h_x = 2\delta\lambda_x$. The proof is given in Appendix A.[1]

As an example, in the chain topology with five nodes depicted in Fig. 2, a transmission from N_2 to N_3 will fail if at the same time slot that N_2 is transmitting, N_3 or N_4 transmits as well. Note that in this topology, where successive nodes are equally far apart, assuming a two-ray ground reflection propagation model with the default capture threshold of 10 dB, a transmission from N_1 or N_5 will not collide with the reception at N_3 due to capture effect [32] (i.e., transmissions from nodes two hops or farther away cannot cause any collision). Therefore, the probability of a successful transmission from N_2 to N_3 equals $p_{2,3} = (1 - 2\delta\lambda_3)(1 - 2\delta\lambda_4)(1 - p_e L_p)$. In fact, the following equation can be used to compute the probability of successful transmission from N_i to N_j, when N_i and N_j are neighbors

$$p_{i,j} = (1 - p_e L_p) \prod_{N_x \in I_j - \{N_i\}} (1 - 2\delta\lambda_x). \tag{4}$$

1. All appendices of this article are available at http://www.cs.mun.ca/winemocol/articles/TMC2017_KafaieAhmedChenDobre_Appendices.pdf.
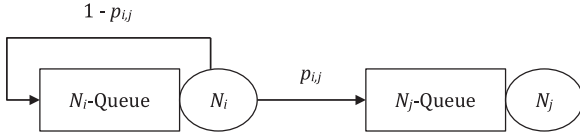
Fig. 3. Feedback queue to model retransmission.

4 PROBLEM FORMULATION

In this section, we first provide the analytical model for traditional forwarding (i.e., non-coding scheme); then we extend it to the case that intermediate nodes can utilize network coding and combine packets of the two flows (i.e., coding scheme).

4.1 Non-Coding Scheme

In the non-coding scheme, the intermediate nodes forward only native packets, while the packets may enter the network (i.e., the queue network) either at node N_1 with a generation rate γ_1 or at node N_k with a generation rate γ_k, and depart from the other end of the chain. Therefore, the intermediate nodes receive packets from both directions. Let $\lambda_i^{(1)}$ and $\lambda_i^{(2)}$ denote the arrival rate of the first flow (i.e., from N_1 to N_k) and the second flow (i.e., from N_k to N_1) arriving at node N_i, respectively. Therefore, at each node $\lambda_i = \lambda_i^{(1)} + \lambda_i^{(2)}$.

We consider each node as a single $M/M/1/\infty$ queuing model. As explained earlier, the departure time distribution in an $M/M/1$ queue with arrival rate λ, in a stable state, is an exponential distribution with mean $1/\lambda$. One of the key rules of probability used in this model states that "the sum of t independent Poisson processes with arrival rates $\lambda_1, \ldots, \lambda_t$ is also a Poisson process with an arrival rate $\lambda = \sum_{i=1}^{t} \lambda_i$" [27]. Hence, the assumption of having Poisson arrivals at intermediate nodes holds, and each node can be considered as an independent $M/M/1$ queuing system.

To model the retransmission of packets in the network, feedback queues are required. As shown in Fig. 3, we consider that node N_i delivers its packets to the next-hop N_j successfully with the probability $p_{i,j}$, and retransmits the packets with the probability $1 - p_{i,j}$, at most $\beta - 1$ times (i.e., the packet is retransmitted if the last transmission fails). Hence, a packet is dropped if it cannot be delivered to the next-hop after β transmissions. This drop probability for a packet sent from N_i to the next-hop N_j, can be calculated as

$$P_{i,j}^d = (1 - p_{i,j})^\beta. \tag{5}$$

Taking retransmissions into account, (6a) and (6b) represent the arrival rate of the first flow (i.e., from N_1 to N_k) and the second flow (i.e., from N_k to N_1) at all nodes, respectively

$$\begin{cases} \lambda_i^{(1)} = \gamma_i + \lambda_i^{(1)}(1 - p_{i,i+1})(1 - P_{i,i+1}^d) & \text{if } i = 1 \\ \lambda_i^{(1)} = \lambda_{i-1}^{(1)} p_{i-1,i} \\ \qquad + \lambda_i^{(1)}(1 - p_{i,i+1})(1 - P_{i,i+1}^d) & \text{if } 1 < i < k \\ \lambda_i^{(1)} = \lambda_{i-1}^{(1)} p_{i-1,i} & \text{if } i = k \end{cases} \tag{6a}$$

$$\begin{cases} \lambda_i^{(2)} = \gamma_i + \lambda_i^{(2)}(1 - p_{i,i-1})(1 - P_{i,i-1}^d) & \text{if } i = k \\ \lambda_i^{(2)} = \lambda_{i+1}^{(2)} p_{i+1,i} \\ \qquad + \lambda_i^{(2)}(1 - p_{i,i-1})(1 - P_{i,i-1}^d) & \text{if } 1 < i < k \\ \lambda_i^{(2)} = \lambda_{i+1}^{(2)} p_{i+1,i} & \text{if } i = 1. \end{cases} \tag{6b}$$

4.1.1 Successful Transmission Probabilities

As explained in Section 3.3, the probability of transmitting a packet successfully can be calculated in terms of the packet arrival rates and propagation delay by solving the following system of non-linear equations:

$$\begin{cases} p_{1,2} = (1 - p_e L_p)(1 - 2\delta\lambda_2)(1 - 2\delta\lambda_3) \\ \cdots \\ p_{i-1,i} = (1 - p_e L_p)(1 - 2\delta\lambda_i)(1 - 2\delta\lambda_{i+1}) \\ \cdots \\ p_{k-2,k-1} = (1 - p_e L_p)(1 - 2\delta\lambda_{k-1})(1 - 2\delta\lambda_k^{(2)}) \\ p_{k-1,k} = (1 - p_e L_p)(1 - 2\delta\lambda_k^{(2)}) \\ p_{k,k-1} = (1 - p_e L_p)(1 - 2\delta\lambda_{k-1})(1 - 2\delta\lambda_{k-2}) \\ \cdots \\ p_{i+1,i} = (1 - p_e L_p)(1 - 2\delta\lambda_i)(1 - 2\delta\lambda_{i-1}) \\ \cdots \\ p_{3,2} = (1 - p_e L_p)(1 - 2\delta\lambda_2)(1 - 2\delta\lambda_1^{(1)}) \\ p_{2,1} = (1 - p_e L_p)(1 - 2\delta\lambda_1^{(1)}), \end{cases} \tag{7}$$

where all λ_is are functions of γ_1, γ_k, and successful transmission probabilities as described in (6).

4.1.2 Service Time

The average service time (i.e., $1/\mu$), which is the time until a packet at the head of the transmission queue of N_i is delivered to the next-hop N_j, can be computed as

$$\frac{1}{\mu} = \sum_{m=1}^{\beta} p_{i,j}(1 - p_{i,j})^{m-1} \sum_{n=1}^{m} T_s(n), \tag{8}$$

where $T_s(m)$ denotes the service time at the mth transmission of a packet, which is presented by $T_s(m) = T(m) + T_{\text{data}} + \delta + \text{SIFS} + T_{\text{ack}} + \delta, 1 \leq m \leq \beta$. T_{data} is the transmission time of a packet (we assume the length of the packets is fixed.), T_{ack} denotes the transmission time of an acknowledgement, and $T(m)$ is calculated for the mth transmission of a packet in terms of DIFS and back-off time, considering the "clock freezing" feature.

To explain "clock freezing", let us use the scenario depicted in Fig. 4. As shown in this figure, node N_i sets the timer for T_{counter} to back-off before transmitting its packet. However at t_1, before the back-off timer reaches zero, N_i senses a packet transmission from N_j, stops counting down, and freezes the state for $T_{\text{trans}} = T_{\text{data}} + T_{\text{ack}} + \text{SIFS}$. Then, since another transmission by N_k occurs, N_i needs to wait until t_3. After that N_i senses the idle channel, resumes the timer, and it is ready to transmit the packet at t_4. Note that to consider *network allocation vector (NAV) virtual*
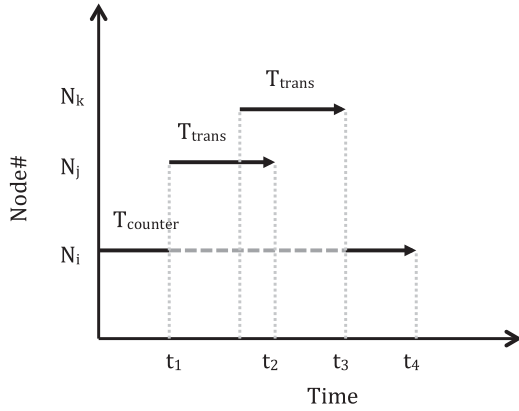
Fig. 4. Clock freezing behavior of the back-off timer.

carrier-sensing mechanism, we take into account T_{ack} and SIFS in calculating T_{trans}.

To take into account this "clock freezing" behavior, we compute the waiting time due to DIFS and back-off as follows:

$$T(m) = T_{counter}(m) \times e^{-\lambda T_{counter}(m)}$$
$$+ \sum_{i=1}^{\infty} (T_{counter}(m) + i \times T_{trans}) \times (1 - e^{-\lambda T_{trans}})^i, \quad (9)$$

which means that a node waits $T_{counter}$ with the probability that during this period of time, it does not sense any other transmission. In addition, a node waits for $T_{counter} + i \times T_{trans}$ with the probability that during each T_{trans} time period, the node senses at least one packet transmission, and it happens i times. This equation provides an upper-bound for the expected back-off time. Its closed form is calculated as follows, and the proof is given in Appendix B.

$$T(m) = T_{counter}(m) \times e^{-\lambda T_{counter}(m)}$$
$$+ (1 - e^{-\lambda T_{trans}}) \left(\frac{T_{counter}(m)}{e^{-\lambda T_{trans}}} + \frac{T_{trans}}{e^{-2\lambda T_{trans}}} \right). \quad (10)$$

In (9) and (10), λ represents the sum of arrival rates of the group of nodes which are in carrier sensing range of this node, and the term in the second line of both equations presents the probability of i transmissions from the nodes of this group during the waiting time $T_{counter}(m)$. In addition, $T_{counter}(m)$ is calculated in terms of m, the number of transmissions of a packet, considering the *binary exponential random back-off* interval, as

$$T_{counter}(m) = \text{DIFS} + \frac{2^{m-1}CW_{min} - 1}{2}, 1 \le m \le \beta. \quad (11)$$

Note that since the random back-off has a uniform distribution, its mean for the mth transmission equals $\frac{2^{m-1}CW_{min}-1}{2}$.

4.1.3 Throughput

It is clear that the throughput, denoted by θ, is identical to the arrival rate at the destinations. Thus, it can be calculated by adding the arrival rate of the second flow at N_1 and the arrival rate of the first flow at N_k as follows:

$$\theta = \lambda_1^{(2)} + \lambda_k^{(1)}. \quad (12)$$

4.1.4 End-to-End Delay

The average end-to-end delay equals the summation of the time that each packet spends at the source and intermediate nodes. Also, the time spent at each node consists of the waiting time in the queue, and the time which takes a packet at the head of the queue to be delivered to the next-hop (i.e., service time). Based on queuing theory, the average time a packet spends at node N_i until it is delivered to the next-hop, defined as W_i, can be expressed as

$$W_i = \frac{1}{\mu_i - \lambda_i}. \quad (13)$$

Since in (8) we calculated an upper-bound of the service time (i.e., an upper-bound of $T(m)$), W_i presents an upper-bound of the waiting time at node N_i. There are two flows in the network; hence, we calculate the end-to-end delay for the packets of each flow separately, and then we compute the average end-to-end delay by applying the weighted average over the end-to-end delay of the two flows. It is clear that the end-to-end delay for each flow equals the sum of waiting time of the packets of the flow in different nodes, except for the destination. Therefore, an upper-bound of the end-to-end delay for the first and second flows can be computed by (14a) and (14b), respectively

$$W^{(1)} = W_1^{(1)} + \sum_{i=2}^{k-1} W_i \quad (14a)$$

$$W^{(2)} = W_k^{(2)} + \sum_{i=2}^{k-1} W_i, \quad (14b)$$

where $W_i = \frac{1}{\mu_i - (\lambda_i^{(1)} + \lambda_i^{(2)})}$ for intermediate nodes.

Note that while at intermediate nodes the packets of both flows arrive in the queue, in the sources' queue the only packets arrived are those of the flow initiated from that node. Due to this reason, the waiting times at the sources are $W_1^{(1)} = \frac{1}{\mu_1 - \lambda_1^{(1)}}$ and $W_k^{(2)} = \frac{1}{\mu_k - \lambda_k^{(2)}}$. Then, an upper-bound of the average end-to-end delay can be computed as

$$W = \frac{\gamma_1}{\gamma_1 + \gamma_k} \times W^{(1)} + \frac{\gamma_k}{\gamma_1 + \gamma_k} \times W^{(2)}. \quad (15)$$

4.2 Coding Scheme

To model network coding, we use multi-class queuing networks, and consider that native and coded packets enter separate queues. Furthermore, coded packets in Q^c have a non-preemptive higher priority over the native packets in Q^n. This means that a coded packet will be forwarded earlier than all the packets waiting in Q^n, but a native packet in service (i.e., the native packet which is being transmitted) is not interrupted by coded packets.

As in the previous case, we assume that the rate of generating packets at N_1 and N_k equals γ_1 and γ_k, respectively, and $\lambda_i^{(1)}$ and $\lambda_i^{(2)}$ represent the rate of the first and the second flow at N_i, respectively. Also, we define λ_i^n as the
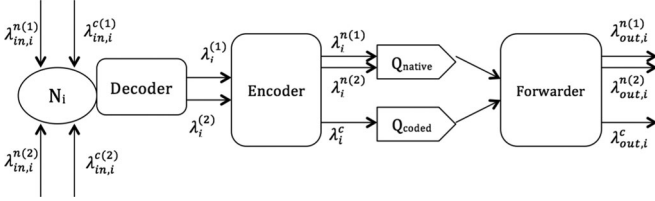
Fig. 5. A packet from arrival until departure.

arrival rate of native packets, and λ_i^c as the arrival rate of coded packets at N_i.

4.2.1 Coding Module

As shown in Fig. 5, N_i receives native and coded packets of both flows from the previous hops. Although a coded packet is the combination of both flows, the receiver N_i is the next-hop of either the first flow or the second flow (i.e., intended flow). Due to this reason, we distinguish coded packets of different flows arriving at N_i.

The *decoder*, in Fig. 5, decodes the received coded packets and finds the next-hop of the packets. The outputs of this module are native packets of the first and the second flows with rates $\lambda_i^{(1)}$ and $\lambda_i^{(2)}$, respectively. In fact $\lambda_i^{(1)}$ ($\lambda_i^{(2)}$) represents the sum of the arrived native packets of the first (second) flow, denoted by $\lambda_{in,i}^{n(1)}$ ($\lambda_{in,i}^{n(2)}$), the successfully decoded packets of the first (second) flow, and retransmitted packets. Therefore, the arrival rates at the *encoder* for both flows (i.e., $\lambda_i^{(1)}$ and $\lambda_i^{(2)}$ in Fig. 5) are calculated as

$$
\begin{cases}
\lambda_i^{(1)} = \lambda_{in,i}^{n(1)} + \lambda_{in,i}^{c(1)} P_{i-1,i}^{\text{decode}} \\
\qquad + \lambda_i^{(1)}(1 - p_{i,i+1})(1 - P_{i,i+1}^d) & \text{if } 1 \le i < k \\
\lambda_i^{(1)} = \lambda_{in,i}^{n(1)} + \lambda_{in,i}^{c(1)} & \text{if } i = k
\end{cases}
$$
(16a)

$$
\begin{cases}
\lambda_i^{(2)} = \lambda_{in,i}^{n(2)} + \lambda_{in,i}^{c(2)} P_{i+1,i}^{\text{decode}} \\
\qquad + \lambda_i^{(2)}(1 - p_{i,i-1})(1 - P_{i,i-1}^d) & \text{if } 1 < i \le k \\
\lambda_i^{(2)} = \lambda_{in,i}^{n(2)} + \lambda_{in,i}^{c(2)} & \text{if } i = 1.
\end{cases}
$$
(16b)

Note that in a general topology, to decode a coded packet with two coding partners, the node should have already received one of them from the opposite direction. Therefore, the decoding probability of a coded packet arrived at N_i from N_{i-1} (or N_{i+1}) is $P_{i-1,i}^{\text{decode}} = (1 - P_{i+1,i}^d)$ (or $P_{i+1,i}^{\text{decode}} = (1 - P_{i-1,i}^d)$). However, in the chain topology discussed in this paper, the decoding probability is always one (i.e., $P_{i-1,i}^{\text{decode}} = P_{i+1,i}^{\text{decode}} = 1$). In fact, if N_i receives coded packet $P_1 \oplus P_2$ from N_{i-1}, and P_1 is its intended packet (i.e., the packet that this node was its next-hop), it must have already received P_2 from N_{i+1}; otherwise, N_{i-1} could not have received P_2 to combine it with P_1.

Previous analytical studies on network coding usually do not consider opportunistic coding, and assume that the transmission of a native packet at the head of the queue, ready to be forwarded, is postponed until receiving packets from other flows, to mix them with the native packet, and send coded packets instead of native ones as much as possible. This assumption provides more coding opportunities, and simplifies estimating the rate of coding opportunities

(i.e., forwarding coded packets) at each node. For example, in the chain topology explained here, the rate would be calculated as the minimum of the arrival rates of the flows.

However, this postponing will increase the end-to-end delay extremely, especially when the flows are asymmetric as the transmission of native packets should be delayed, waiting for coding partners to arrive. In addition, many practical and well-known network coding protocols are designed based on opportunistic coding, and do not impose such an artificial delay [22], [33], [34]. To limit the delay in the network, and also to analyze the behavior of network coding in practical scenarios, we do not hold transmission of native packets. This means that the arrival rate in Q^c is not the minimum of the arrival rates of the two flows any more, and can be calculated as will be explained here.

In our model, a packet may be transmitted natively if it is at the head of Q^n, and there is no packet in Q^c. Therefore, the encoder receives the arrived native packet P from flow r ($r = 1, 2$), and looks for a packet from flow $\bar{r}$ (i.e., the flow from the opposite direction that can be mixed with flow r, $\bar{r} = 3 - r$) in Q^n. If the node finds such a packet $\bar{P}$, it removes $\bar{P}$ from Q^n, mixes it with P and adds the coded packet to Q^c; otherwise, it will add P to Q^n. Therefore, a packet will be added to Q^c if a native packet from flow r arrives at the encoder, and if Q^n contains at least one packet from the flow $\bar{r}$.

On the other hand, packet P, from flow r, will be sent natively if before it is forwarded, it cannot be mixed with any packet from the other flow. This happens if 1) when it arrives, the queue of the other flow is empty, and 2) during the time that P is waiting in Q^n to be forwarded, the number of packets of the other flow which arrive in Q^n is less than the number of packets of flow r in Q^n ahead of P. Note that although all native packets arrive at the same queue, we send them to two separate virtual queues, one for each flow, to be able to calculate the number of packets of each flow in the queue.

If we denote $W(Q^n)$ as the waiting time of an arrived native packet in Q^n, then the number of packets of flow r that arrive in Q^n during this time equals $N(r, W(Q^n)) = \lambda_i^{(r)} \times W(Q^n)$. When P from flow r arrives, if the number of packets of its flow in Q^n (i.e., packets of flow r ahead of P) is less than $N(\bar{r}, W(Q^n))$, P is moved from Q^n to Q^c before it is forwarded; otherwise, it stays in Q^n. Thus, the probability that a packet from flow r moves to the coded queue of node N_i, Q_i^c, even if it first arrives in the native queue, Q_i^n, can be calculated as

$$
\begin{aligned}
P_{\text{mtc}}(r) &= \Pr[N(\bar{r}, W(Q^n)) > N(r)] \\
&= \sum_{k=0}^{\infty} \Pr[(N(\bar{r}, W(Q^n)) > N(r)) | (N(r) = k)] \\
&\quad \times \Pr[N(r) = k] \\
&= \sum_{k=0}^{\infty} \Pr[(N(\bar{r}, W(Q^n)) > k)] \Pr[N(r) = k] \\
&= \sum_{k=0}^{\infty} \left(1 - \sum_{j=0}^{k} \frac{e^{\left(-\lambda_i^{n(\bar{r})} W(Q_i^n)\right)} \left(\lambda_i^{n(\bar{r})} W(Q_i^n)\right)^j}{j!} \right) \\
&\quad \left(\frac{\lambda_i^{n(r)}}{\mu_i^{n,\text{seen}}}\right)^k \left(1 - \frac{\lambda_i^{n(r)}}{\mu_i^{n,\text{seen}}}\right),
\end{aligned}
$$
(17)

TABLE 3
Input Rates of Native Packets at All Nodes

i	$\lambda_{in,i}^{n(1)}$	$\lambda_{in,i}^{n(2)}$
$i = 1$	γ_1	$\lambda_{out,i+1}^{n(2)} \times p_{i+1,i}$
$1 < i < k$	$\lambda_{out,i-1}^{n(1)} \times p_{i-1,i}$	$\lambda_{out,i+1}^{n(2)} \times p_{i+1,i}$
$i = k$	$\lambda_{out,i-1}^{n(1)} \times p_{i-1,i}$	γ_k

TABLE 4
Input Rates of Coded Packets at All Nodes

i	$\lambda_{in,i}^{c(1)}$	$\lambda_{in,i}^{c(2)}$
$i = 1, i = 2$	0	$\lambda_{out,i+1}^{c} \times p_{i+1,i}$
$2 < i < k-1$	$\lambda_{out,i-1}^{c} \times p_{i-1,i}$	$\lambda_{out,i+1}^{c} \times p_{i+1,i}$
$i = k, i = k-1$	$\lambda_{out,i-1}^{c} \times p_{i-1,i}$	0

where $N(r, w)$ denotes the number of packets of flow r arrived in Q^n during time window w and $N(r)$ is the number of the packets of flow r ahead of the currently arrived packet in Q^n. Also, $\mu_i^{n,\text{seen}}$ denotes the service time seen by Q_i^n as is discussed later. The closed form of (17) can be computed as

$$P_{\text{mtc}}(r) = 1 - e^{\left(W(Q_i^n)\lambda_i^{n(\bar{r})}\left(\frac{\lambda_i^{n(r)}}{\mu_i^{n,\text{seen}}} - 1 \right) \right)}. \tag{18}$$

We provide the proof in Appendix C.

Next the arrival rate of the packets of the first and second flows in the native queue of N_i is calculated as

$$\lambda_i^{n(1)} = \lambda_i^{(1)} \times \pi_0(Q_i^{n(2)}) \times (1 - P_{\text{mtc}}(1)), \tag{19a}$$

$$\lambda_i^{n(2)} = \lambda_i^{(2)} \times \pi_0(Q^{n(1)}) \times (1 - P_{\text{mtc}}(2)), \tag{19b}$$

where $\pi_0(Q)$ is the probability that queue Q is empty. This equation means that the arrival rate of native packets of the first flow in Q_i^n (i.e., $\lambda_i^{n(1)}$) equals the arrival rate of the packets of the first flow at the encoder (i.e., $\lambda_i^{(1)}$) for which, in their arrival time, 1) there is no packet from the second flow in Q_i^n (i.e., $\pi_0(Q_i^{n(2)})$), and 2) the packet will stay in Q_i^n during its waiting time in the queue (i.e., $1 - P_{\text{mtc}}(1)$). $\lambda_i^{n(2)}$ is calculated in a similar way. Also, the arrival rate in the coded queue of N_i, can be calculated as

$$\lambda_i^c = \frac{\lambda_i^{(1)} + \lambda_i^{(2)} - \lambda_i^{n(1)} - \lambda_i^{n(2)}}{2}. \tag{20}$$

The division by two is because each coded packet is a combination of two native packets.

4.2.2 Native and Coded Queues

The arrival rates in Q_i^n and Q_i^c equal $\lambda_i^{n(1)} + \lambda_i^{n(2)}$ and λ_i^c, respectively. The *forwarder* module, in Fig. 5, is responsible for forwarding packets. If Q_i^c is not empty, it will select the packet from the head of Q_i^c; otherwise, the packet is chosen from the head of Q_i^n if it is not empty.

As stated earlier, priority queues are used to model this case, where the arrival rate in Q_i^n is the sum of the arrival rates of both flows (i.e., $\lambda_i^n = \lambda_i^{n(1)} + \lambda_i^{n(2)}$), and the total arrival rate in the queuing system of N_i is presented by $\lambda_i = \lambda_i^n + \lambda_i^c$. Knowing the input rate of native and coded packets at all nodes, one can calculate the output rate at different nodes. Note that since we assume that the queuing system is in a stable state, the departure rates equal the arrival rates ($\lambda_{out,i}^{n(1)} = \lambda_i^{n(1)}, \lambda_{out,i}^{n(2)} = \lambda_i^{n(2)}, \lambda_{out,i}^c = \lambda_i^c$). Finally, the throughput can be computed using (12).

Tables 3 and 4 provide the input rates of native and coded packets at all nodes. Moreover, it is clear that the output rate of the first flow at N_1 and that of the second flow at N_k are γ_1 and γ_k, respectively. In addition, the output rate of the second flow and coded packets at N_1 and the output rate of the first flow and coded packets at N_k are equal to zero, as in

$$\begin{cases} \lambda_{out,1}^{n(1)} = \gamma_1 \\ \lambda_{out,k}^{n(2)} = \gamma_k \\ \lambda_{out,1}^{n(2)} = 0 \\ \lambda_{out,k}^{n(1)} = 0 \\ \lambda_{out,i}^c = 0 \quad \text{if } i = 1, k \end{cases} \tag{21}$$

4.2.3 Service Time and End-to-End Delay

As stated earlier, we use two different types of queues for native and coded packets, while the coded packets in Q^c have a non-preemptive higher priority over native packets in Q^n. In such a scenario, the service time seen by the native packets is different from the service time of a regular $M/M/1$ queue. The reason is that a native packet at the head of Q^n should wait for all packets in Q^c to be transmitted before its turn for transmission. To estimate the service time *seen* by the native packets (i.e., the packets in lower priority queue) at N_i, denoted by $\mu_i^{n,\text{seen}}$, we start from the formula in queuing theory, which calculates the waiting time of a packet in a $M/M/1$ queuing system as

$$W_{\text{system}} = \frac{1}{\mu - \lambda}. \tag{22}$$

Therefore, the service time can be calculated as $\mu = \lambda + 1/W_{\text{system}}$. Since for the native queue at N_i, $\lambda = \lambda_i^{n(1)} + \lambda_i^{n(2)}$, the waiting time of the packets in the lower priority queue (i.e., Q_i^n) can be computed as $W(Q_i^n) = \bar{R}_i/(1 - \rho_i^c)(1 - \rho_i^c - \rho_i^n)$, and the waiting time of native packets before delivery to the next-hop equals $W_{\text{system}} = W(Q_i^n) + \frac{1}{\mu_i^n}$, we can calculate the service time *seen* by the packets in Q_i^n as

$$\mu_i^{n,\text{seen}} = \lambda_i^{n(1)} + \lambda_i^{n(2)} + \frac{1}{\frac{\bar{R}_i}{(1-\rho_i^c)(1-\rho_i^c-\rho_i^n)} + \frac{1}{\mu_i^n}}, \tag{23}$$

where μ_i^n is the service time of native packets in a regular queuing system that has been calculated earlier in (8). As presented in (17) and (18), $\mu_i^{n,\text{seen}}$ is used to calculate P_{mtc}. Table 5 shows the required equations to compute variables described in this section.

Furthermore, when a node sends a coded packet, it needs to wait for more than one ACK. In our model with two

TABLE 5
The Calculation of Some Variables' Values

Variable	Equation
ρ_i	$\frac{\lambda_i}{\mu_i}$
$\bar{R}_i$	$\frac{\rho_i^n}{\mu_i^n} + \frac{\rho_i^c}{\mu_i^c}$
$W(Q_i^c)$	$\frac{\bar{R}_i}{(1-\rho_i^c)}$
$W(Q_i^n)$	$\frac{\bar{R}_i}{(1-\rho_i^c)(1-\rho_i^c-\rho_i^n)}$
$\mu_i^{n,seen}$	$\lambda_i^{n(1)} + \lambda_i^{n(2)} + \frac{1}{W(Q_i^n)+\frac{1}{\mu_i^n}}$

flows, the service time for coded packets, μ_i^c, is calculated using (8) again, where

$$T_s(m) = T(m) + T_{\text{data}} + \delta + 2 \times (\text{SIFS} + T_{\text{ack}} + \delta). \quad (24)$$

Since packets in Q^c and Q^n have different average waiting times, we calculate the waiting time of native and coded packets separately at each node, and then apply the weighted average to compute the average waiting time at each node, as

$$W_i = \frac{\lambda_i^c}{\lambda_i^c + \lambda_i^n} \times \left(W(Q_i^c) + \frac{1}{\mu_i^c} \right) + \frac{\lambda_i^n}{\lambda_i^c + \lambda_i^n} \times \left(W(Q_i^n) + \frac{1}{\mu_i^n} \right). \quad (25)$$

Finally, the average end-to-end delay can be computed using (14) and (15). Note that we assume that the encoding and decoding delays are negligible, and the coding overhead is small enough that we can consider similar length for coded and native packets. In addition, since we calculate an upper-bound of $T_s(m)$, our analytical model provides an upper-bound of the end-to-end delay for both non-coding and coding schemes.

5 PERFORMANCE EVALUATION

5.1 Network Description

To verify the accuracy of our proposed analytical model, we run simulations in NS-2 for the chain topology depicted in Fig. 1, where the distance between successive nodes is

200 m, and N_1 and N_k transmit packets to each other via intermediate nodes $N_2, \ldots, N_{k-1}$. The channel propagation used in NS-2 is a two-ray ground reflection model [32], the transmission range is 250 m, and the carrier sensing range is 550 m. Hence, in our chain topology, the nodes within two-hop distance of each node are in its carrier sensing range. However, due to the capture effect, the interference range is limited to the nodes one hop away.

In our simulation, we use the IEEE 802.11 standard as the MAC layer protocol, and our physical layer introduces random packet loss by adopting bit error rates (p_e). Therefore, the receiver will drop the packet with a probability which is calculated in terms of p_e. In addition, a node may drop a packet due to collision. Based on the specifications, a node transmits a packet at most 7 times (i.e., $\beta = 7$).

The link rate is set to 2 Mbps. The sources, in our simulation scenarios, send Poisson data flows with a datagram size of 1,000 bytes. We compare the analytical results with the simulation results in different scenarios in terms of throughput and end-to-end delay by varying the packet generation rate and bit error rate. Also, to compare coding and non-coding schemes, we calculate the maximum stable throughput for both cases.

5.2 The Effect of Packet Generation Rate

In this section, we compare the simulation and analytical results for several packet generation rates in the topology depicted in Fig. 2 with five nodes. In our simulations, Poisson flows between N_1 and N_5 last for 170 seconds. We change the generation rate of packets at sources while the bit error rate is fixed to 2×10^{-6}, and calculate the total throughput and an upper-bound of the average end-to-end delay by assuming an equal packet generation rate at sources (i.e., $\gamma = \gamma_1 = \gamma_k$). We compare the simulation and analytical results for the cases that 1) nodes do not retransmit a packet even if its transmission fails, and 2) nodes transmit a packet at most β times ($\beta = 7$).

Fig. 6a presents the analytical and simulation results of throughput for non-coding scheme, both with and without retransmission. Also, Fig. 6b shows the same result for coding scheme. The consistency of the simulation and analytical results corroborates the validity of our analytical model.
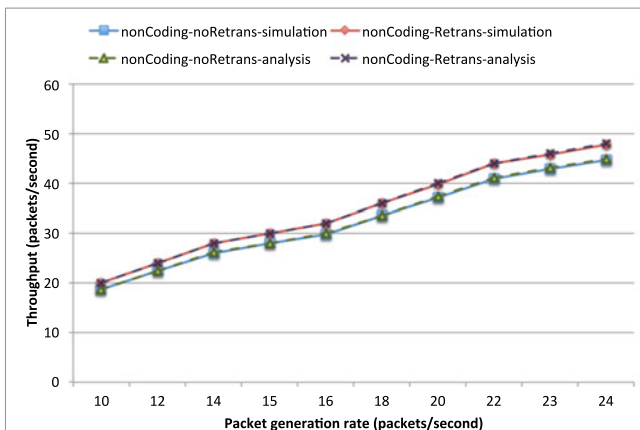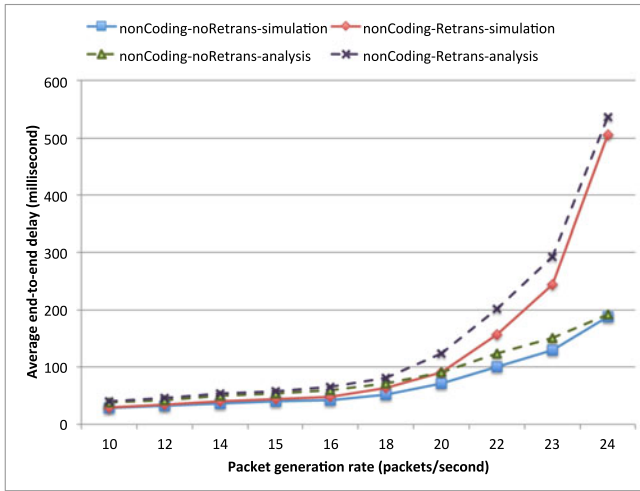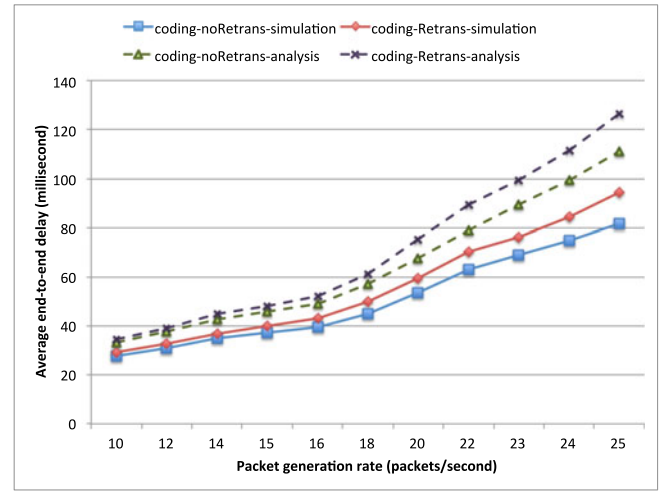

(a) Non-coding scheme.
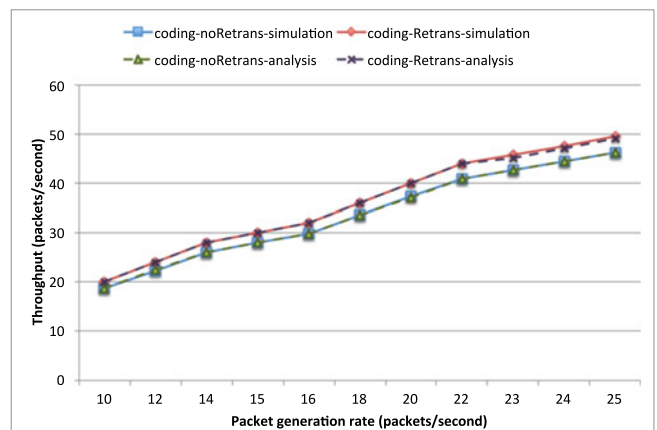
(b) Coding scheme.

Fig. 6. Throughput comparison for different packet generation rates in a chain topology with five nodes and $p_e = 2 \times 10^{-6}$.

(a) Non-coding scheme.

(b) Coding scheme.

Fig. 7. The average end-to-end delay comparison for different packet generation rates in a chain topology with five nodes and $p_e = 2 \times 10^{-6}$.

In addition, one may notice that the throughput at each given packet generation rate is higher when retransmission is enabled. In fact, by disabling the reransmission mechanism, all the efforts to deliver a packet are wasted even if it has made all the way but the very last hop.

Comparing Figs. 6a and 6b, no considerable throughput gain can be seen for coding scheme in comparison with non-coding scheme especially in lower arrival rates. This is due to the fact that without holding native packets, network coding usually shows its gain over the traditional forwarding approach, where arrival rates are high enough to provide frequent coding opportunities. We will discuss the gain further in Section 5.4.

Regarding the average end-to-end delay, the results in Fig. 7 show that our analytical model provides an upper-bound for the average end-to-end delay in different packet generation rates for both non-coding and coding schemes. In addition, in both scenarios (i.e., with and without retransmission), the average end-to-end delay increases with the packet generation rate; the reason is that at higher generation rates more packets are queued at nodes, which increases the waiting time and consequently the end-to-end delay of the network. However, the end-to-end delay is shorter when retransmission is disabled because each packet has only one transmission chance to be delivered to the next-hop, and lost packets do not contribute to delay calculation.

As a matter of fact, without retransmission a packet is either dropped or delivered to the next hop with only one transmission. On the other hand, with enabling retransmission, the packet is provided with up to β chances to repeat, which improves throughput at the cost of a longer delay. Furthermore, as shown in Fig. 7, without utilizing network coding the delay grows faster. This is due to the fact that network coding allows more than one packet to be delivered to the next-hop in one transmission, which accelerates packet delivery, and reduces contention.

5.2.1 Throughput-Delay Trade-Off

As presented in Fig. 6, if the end-to-end delay of the network is finite (i.e., the queues are in stable state), the throughput is an increasing function of packet generation rate [23]. On the other hand, Fig. 7 shows that the end-to-end delay is also an increasing function of the packet generation rate. As explained earlier, this is due to the fact that generating new packets faster increases the number of packets queued to be transmitted, which means longer waiting time in the queues, as confirmed by (13).

This verifies a trade-off between throughput and end-to-end delay that has been discussed in the literature [8], [23], [35]. To find the delay-constraint capacity, one needs to calculate the optimal packet generation rate for a given end-to-end delay. In our model, for both traditional forwarding and network coding, it can be calculated by increasing the packet generation rate as long as the end-to-end delay is less than the given value. Doing so, one can obtain the packet generation rate in which the network achieves the maximum throughput satisfying the end-to-end delay requirement.

5.3 The Effect of Bit Error Rate

To study the validity of our model under different link qualities and packet loss probabilities, we change the bit error rate, and provide simulation and analytical results for coding and non-coding schemes for the same topology depicted in Fig. 2, both with and without retransmission. In these experiments, the packet generation rate at both sources (i.e., N_1 and N_5) is set to 20 packets/second. In general, as shown in Figs. 8 and 9, at lower bit error rates, the network performance with retransmission is very close to the case that retransmission is disabled. This is because at higher link qualities most of the packets are delivered to the next hop with one transmission without any need to retransmission.

As shown in Fig. 8, the throughput calculated based on the proposed model perfectly matches the simulation results for different bit error rates. In addition, when retransmission is disabled, the throughput drops with increasing the bit error rate. On the other hand by enabling retransmission, the throughput remains almost constant especially for the non-coding scheme. The reason is that retransmission provides each packet with up to β chances to be delivered to the next-hop, which is usually sufficient
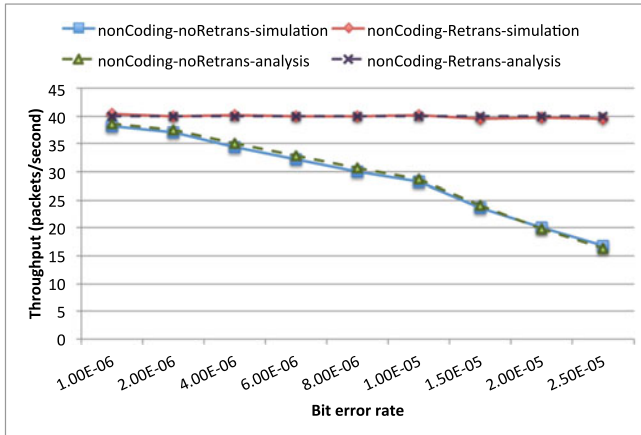
(a) Non-coding scheme.
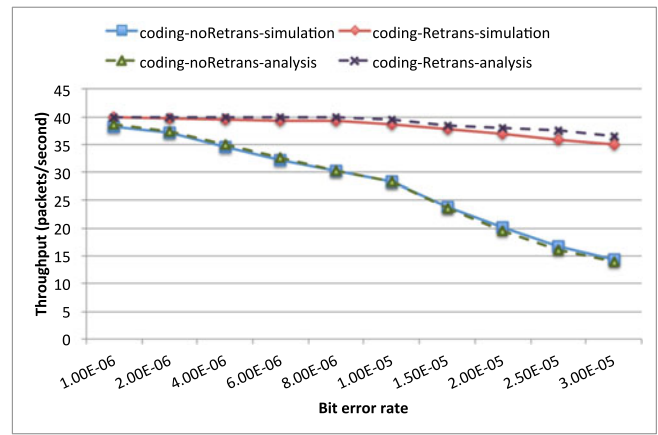
(b) Coding scheme.

Fig. 8. Throughput comparison for different bit error rates in a chain topology with five nodes and $\gamma = 20$.

for most packets in these scenarios even at higher bit error rates. One may notice that the coding scheme does not seem as resilient as the non-coding scheme when retransmission is enabled; the reason is that, to decode each coded packet, two packets should be delivered successfully rather than one, which reduces the chance of successful delivery of coded packets even when retransmission is enabled.

Regarding the average end-to-end delay, as shown in Fig. 9, when the retransmission is disabled, the delay decreases for higher bit error rates. The reason is that more packets are dropped, and dropped packets do not contribute to the delay calculations. In addition, by increasing the packet loss rate, the number of packets waiting in the transmission queue of nodes decreases, which again causes a shorter end-to-end delay for delivered packets. On the other hand, the delay increases with the bit error rate when the retransmission mechanism is utilized, as packets require more retransmissions to get to the next-hop; this adds to both service time and waiting time.

Comparing the coding and non-coding schemes, the effect of the bit error rate is less on coding scheme than on non-coding scheme because in the coding scheme more packets can be forwarded in each transmission. Moreover, as shown in Fig. 9, the average end-to-end delay calculated

based on our analytical model provides an upper bound for the simulation results in all scenarios.

5.4 Maximum Stable Throughput

In this section, we compare the maximum stable throughput of the coding and non-coding schemes using both analytical and simulation results. The maximum stable throughput, as the name suggests, presents the maximum throughput of the network while the nodes' queues are still in a stable state (i.e., the arrival rate is less than the service rate). In these experiments, the bit error rate is set to 2×10^{-6}, and the results are provided for the chain topology depicted in Fig. 1 with variant number of nodes.

To find the maximum stable throughput in simulations for each network size, we increase the packet generation rate at the sources as the throughput increases, and the queues are in stable state. In our analytical model, we follow the same idea since the maximum stable throughput is an increasing function of the packet generation rate. We gradually increase the packet generation rate at the sources. For each given generation rate, the system of non-linear equations provided in Section 4 is solved, providing us with the arrival rates at all nodes as well as other required parameters. Then by calculating the service rates, we can verify
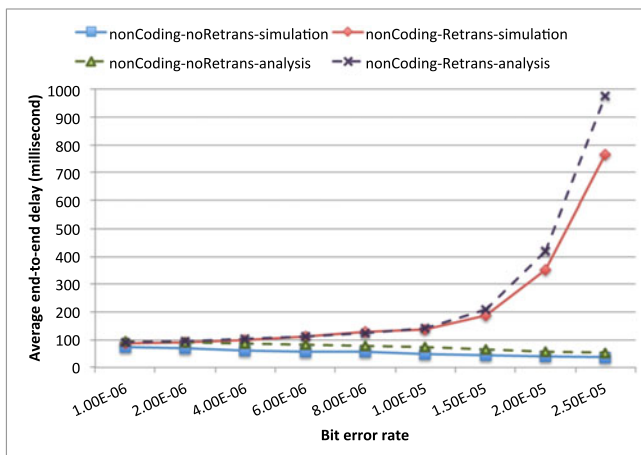
(a) Non-coding scheme.
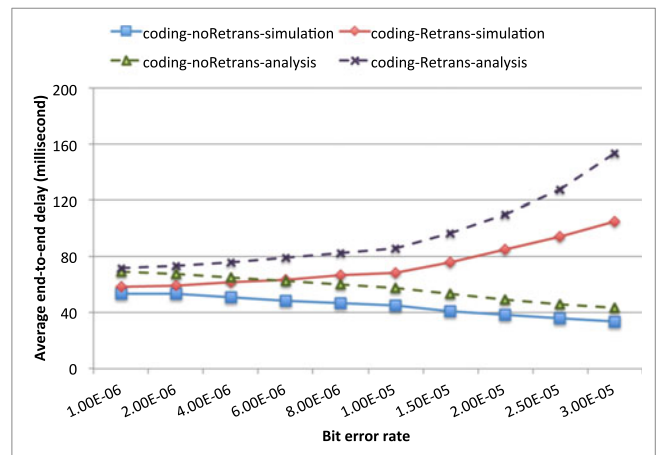
(b) Coding scheme.

Fig. 9. The average end-to-end delay comparison for different bit error rates in a chain topology with five nodes and $\gamma = 20$.

```
Initialization:
γ₁ = γₖ = γini   //arrival rate at the sources
θ = 0            //throughput
for each node i
    ρᵢ < 1

Main Procedure:
While (∀1 < i < k, ρᵢ < 1)
    θmax = θ
    solve the system of non-linear equations and
    calculate θ, λᵢ, 1 < i < k
    for each node i
        calculate μᵢ
        ρᵢ = λᵢ/μᵢ
    γ₁ = γₖ = γ₁ + 1

return θmax
```

Fig. 10. Pseudo-code of calculating the maximum stable throughput. γ_1 and γ_k represent the packet generation rates at the sources, initialized with a small value γ_{ini}. θ denotes the calculated throughput for the given generation rate.

whether all nodes are still in a stable state. As soon as the condition of stability is not valid in at least one node, the packet generation rate is not acceptable at the sources anymore, and we calculate the throughput for the greatest acceptable generation rate as the maximum stable throughput. Fig. 10 presents the pseudo-code for finding the maximum stable throughput in our analytical model.

As shown in Fig. 11, our analytical model provides a good estimate of the maximum stable throughput of the network for both coding and non-coding schemes in chain topologies with different sizes. Fig. 11a presents the results when the retransmission mechanism is disabled, while in Fig. 11b, nodes are allowed to transmit each packet at most β times. In both cases, by increasing the number of nodes in the topology, the maximum stable throughput decreases, especially in smaller topologies. In our chain topology, the number of nodes in the carrier sensing range of a transmitter is between 2 and 4, depending on the transmitter's location. As the chain length increases, a larger fraction of the nodes will have four nodes in their carrier sensing range, which leads to more waiting due to CSMA random access. This causes a longer back-off time and consequently a longer service time, reducing the maximum stable throughput.

Furthermore, as also stated in [18], when the number of intermediate nodes increases, network coding's advantage over traditional routing fades out, and the maximum stable throughput of the coding scheme approaches that of the non-coding scheme. One reason is that most coding opportunities are provided by the middle node, where the arrival rate of packets from both directions are similar and balanced. As the chain topology grows (i.e., the number of hops increases), less packets from both directions can be received by the middle nodes, which reduces the coding opportunities. In addition, in longer chains, the ratio of unbalanced flows increases in other nodes, which further causes less coding opportunities.

6 CONCLUSION AND FUTURE WORK

In this article, we utilized queuing theory to study the throughput and end-to-end delay of both traditional forwarding (i.e., non-coding scheme) and inter-flow network coding in multi-hop wireless mesh networks, where two unicast sessions in opposite directions traverse the network. We proposed an analytical framework considering the specifications of the IEEE 802.11 DCF, such as the binary exponential back-off time with clock freezing and virtual carrier sensing, to formulate the links quality, waiting time of the packets and retransmissions. Our analytical model assumes M/M/1 queues, which are in a stable state, while coded and native packets arrive at separated queues and coded packets have a non-preemptive higher priority over native packets. Furthermore, in our model as opposed to previous studies, the transmission of native packets is not artificially delayed for generating more coded packets; this makes it significantly more challenging to estimate coding opportunities at nodes, as described in Section 4.2.1.

We verified the accuracy of the proposed analytical model by computer simulation in NS-2, and the consistency of the results corroborates the validity of the model. Also, the results show that at any given packet generation rate, both throughput and end-to-end delay are higher when retransmission is enabled. However, when the bit error rate increases, the trend is totally different with and without retransmission. By enabling retransmission, throughput
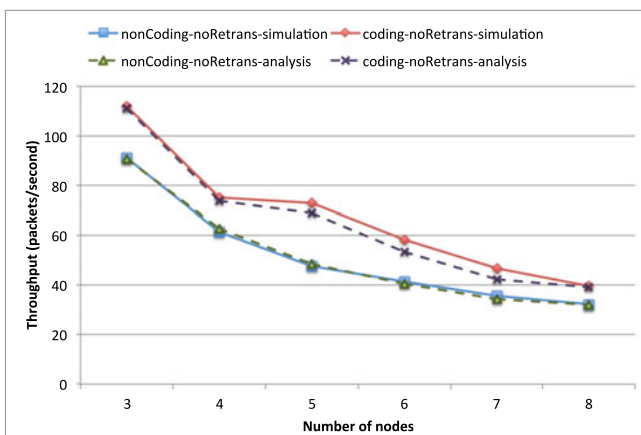
(a) Without retransmission.
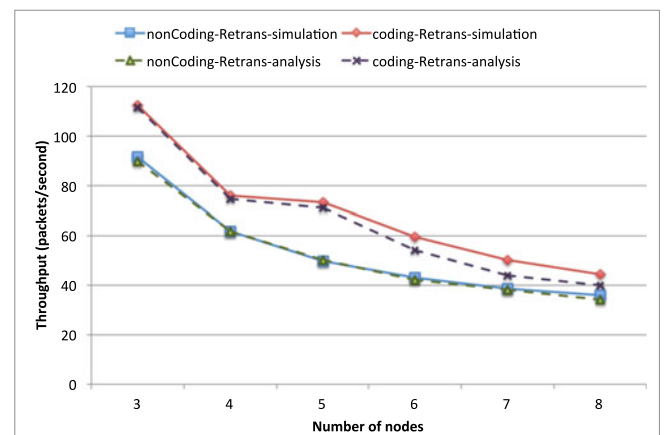
(b) With retransmission.

Fig. 11. The maximum stable throughput comparison for different chain topology sizes, $p_e = 2 \times 10^{-6}$.

stays almost constant across different bit error rates while the end-to-end delay increases significantly. On the other hand, when retransmission is disabled, both throughput and end-to-end delay are decreasing functions of the bit error rate. In addition, while network coding in theory promises a greater capacity for wireless networks, the results for the maximum stable throughput show that when PHY/MAC layer constraints are taken into account, this promise can be fulfilled better for smaller topologies. In fact, when the number of intermediate nodes increases, the maximum stable throughput of network coding becomes comparable to traditional forwarding. However, wireless mesh networks are meant as an extended access technology, and it is unlikely to have very long paths; thus network coding can still offer a competitive edge.

Although our analytical model was formulated in a chain topology, it is applicable to any topology as long as the two opposite flows follow the same path. A future extension of our work could be to develop an analytical framework for a general topology, where more than two flows are traveling and possibly mixing together. In addition, we plan to incorporate cooperative forwarding to our model, where the neighbors of the next-hop can forward the packet if the next-hop itself does not receive it.

Physical-layer network coding (PNC) [36], [37], [38] is a more recent type of network coding, in which nodes simultaneously transmit packets to a relay node that exploits mixed wireless signals to extract a coded packet. In recent years, a number of analytical studies have investigated the throughput capacity of PNC in multi-hop networks [11], [12], [19], and we believe the model proposed here to study the throughput and delay of chain topologies can be extended into PNC, where some two-hop nodes can transmit simultaneously to a relay node without causing collision but these concurrent transmissions increase the carrier sensing range of the network [19].

REFERENCES

[1] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[2] Y. Sagduyu and A. Ephremides, "On broadcast stability region in random access through network coding," in *Proc. 44th Annu. Allerton Conf. Commun. Control Comput.*, 2006, pp. 143–150.

[3] Y. Sagduyu and A. Ephremides, "On network coding for stable multicast communication," in *Proc. IEEE Military Commun. Conf.*, Oct. 2007, pp. 1–7.

[4] N. Moghadam and H. Li, "Queue stability analysis in network coded wireless multicast network," *IEEE Commun. Lett.*, vol. 20, no. 5, pp. 950–953, May 2016.

[5] N. Moghadam and H. Li, "A new wireless multicast queuing design using network coding and data-flow model," *IEEE Commun. Lett.*, vol. 20, no. 8, pp. 1603–1606, Aug. 2016.

[6] G. Paschos, C. Fragiadakis, L. Georgiadis, and L. Tassiulas, "Wireless network coding with partial overhearing information," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 2337–2345.

[7] M. Amerimehr and F. Ashtiani, "Delay and throughput analysis of a two-way opportunistic network coding-based relay network," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2863–2873, May 2014.

[8] V. Jamali, N. Zlatanov, and R. Schober, "Bidirectional buffer-aided relay networks with fixed rate transmission-Part II: Delay-constrained case," *IEEE Trans. Wireless Commun.*, vol. 14, no. 3, pp. 1339–1355, Mar. 2015.

[9] D. Zeng, S. Guo, Y. Xiang, and H. Jin, "On the throughput of two-way relay networks using network coding," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 191–199, Jan. 2014.

[10] J. Le, J. Lui, and D.-M. Chiu, "On the performance bounds of practical wireless network coding," *IEEE Trans. Mobile Comput.*, vol. 9, no. 8, pp. 1134–1146, Aug. 2010.

[11] S. Lin and L. Fu, "Unsaturated throughput analysis of physical-layer network coding based on IEEE 802.11 distributed coordination function," *IEEE Trans. Wireless Commun.*, vol. 12, no. 11, pp. 5544–5556, Nov. 2013.

[12] S. Lin, L. Fu, J. Xie, and X. Wang, "Hybrid network coding for unbalanced slotted ALOHA relay networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 298–313, Jan. 2016.

[13] M. H. Amerimehr, F. Ashtiani, and M. B. Iraji, "An analytical approach for throughput evaluation of wireless network coding," in *Proc. IEEE Int. Conf. Commun.*, 2009, pp. 1696–1700.

[14] D. Umehara, T. Hirano, S. Denno, M. Morikura, and T. Sugiyama, "Wireless network coding in slotted ALOHA with two-hop unbalanced traffic," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 5, pp. 647–661, Jun. 2009.

[15] D. Umehara, S. Denno, M. Morikura, and T. Sugiyama, "Performance analysis of slotted ALOHA and network coding for single-relay multi-user wireless networks," *Ad Hoc Netw.*, vol. 9, no. 2, pp. 164–179, Mar. 2011.

[16] Y. Sagduyu and A. Ephremides, "Cross-layer optimization of MAC and network coding in wireless queueing tandem networks," *IEEE Trans. Inf. Theory*, vol. 54, no. 2, pp. 554–571, Feb. 2008.

[17] M. Iraji, M. Amerimehr, and F. Ashtiani, "A queueing model for wireless tandem network coding," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Apr. 2009, pp. 1–6.

[18] M. Amerimehr, F. Ashtiani, and S. Valaee, "Maximum stable throughput of network-coded multiple broadcast sessions for WirelessTandem random access networks," *IEEE Trans. Mobile Comput.*, vol. 13, no. 6, pp. 1256–1267, Jun. 2014.

[19] S. Lin and L. Fu, "Throughput capacity of IEEE 802.11 many-to/from-one bidirectional networks with physical-layer network coding," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 217–231, Jan. 2016.

[20] J. Hwang and S.-L. Kim, "Cross-layer optimization and network coding in CSMA/CA-based wireless multihop networks," *IEEE/ACM Trans. Netw.*, vol. 19, no. 4, pp. 1028–1042, Aug. 2011.

[21] S. Kafaie, M. H. Ahmed, Y. Chen, and O. A. Dobre, "Throughput analysis of network coding in multi-hop wireless mesh networks using queueing theory," in *Proc. IEEE Global Commun. Conf.*, Dec. 2016, pp. 1–6.

[22] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," *IEEE/ACM Trans. Netw.*, vol. 16, no. 3, pp. 497–510, Jun. 2008.

[23] S. W. Ko and S. L. Kim, "Delay-constrained capacity of the IEEE 802.11 DCF in wireless multihop networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 5, pp. 1105–1115, May 2016.

[24] V. Jamali, N. Zlatanov, and R. Schober, "Bidirectional buffer-aided relay networks with fixed rate transmission-Part I: Delay-unconstrained case," *IEEE Trans. Wireless Commun.*, vol. 14, no. 3, pp. 1323–1338, Mar. 2015.

[25] J. R. Jackson, "Networks of waiting lines," *Operations Res.*, vol. 5, pp. 518–521, 1957.

[26] J. R. Jackson, "Jobshop-like queueing systems," *Manage. Sci.*, vol. 10, pp. 131–142, 1963.

[27] D. Gross and C. M. Harris, *Fundamentals of Queueing Theory*, 3rd ed. Hoboken, NJ, USA: Wiley, 1998.

[28] L. Kleinrock, *Queueing Systems*. John Wiley & Sons, Inc., Theory, vol. 1, 1975.

[29] L. Kleinrock, *Queueing Systems*. John Wiley & Sons, Inc., Computer Applications, vol. 2, 1976.

[30] P. J. Burke, "The output of a queueing system," *Operations Res.*, vol. 4, pp. 699–714, 1956.

[31] *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, LAN MAN Standards Committee of the IEEE Computer Society, ISO/IEC Standard 8802.11:1999, 1999.

[32] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice Hall, Dec. 2001.

[33] J. Le, J. C. S. Lui, and D. M. Chiu, "DCAR: Distributed coding-aware routing in wireless networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 4, pp. 596–608, Apr. 2010.

[34] J. Zhang, Y. P. Chen, and I. Marsic, "MAC-layer proactive mixing for network coding in multi-hop wireless networks," *Comput. Netw.*, vol. 54, no. 2, pp. 196–207, Feb. 2010.

[35] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah, "Optimal throughput-delay scaling in wireless networks - Part I: The fluid model," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2568–2592, Jun. 2006.

[36] S. Zhang, S. C. Liew, and P. P. Lam, "Hot topic: Physical-layer network coding," in *Proc. 12th Annu. Int. Conf. Mobile Comput. Netw.*, 2006, pp. 358–365.
[37] P. Popovski and H. Yomo, "The anti-packets can increase the achievable throughput of a wireless multi-hop network," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2006, pp. 3885–3890.
[38] S. C. Liew, S. Zhang, and L. Lu, "Physical-layer network coding: Tutorial, survey, and beyond," *Phys. Commun.*, vol. 6, pp. 4–42, 2013.

Somayeh Kafaie received the BSc degree in computer engineering (software) from the Amirkabir University of Technology, Iran, in 2007 and the MSc degree in computer engineering (software) from the Iran University of Science and Technology, Iran, in 2011. She is currently working toward the PhD degree in the Faculty of Engineering and Applied Science, Memorial University of Newfoundland. Her research interests include wireless mesh networks, network coding, opportunistic routing, complex networks, and graph theory. She is a student member of the IEEE.

Mohamed Hossam Ahmed received the PhD degree in electrical engineering from Carleton University, Ottawa, in 2001, where he worked from 2001 to 2003 as a senior research associate. In 2003, he joined the Faculty of Engineering and Applied Science, Memorial University, where he works currently as a full professor. He published more than 135 papers in international journals and conferences. He serves as an editor of the *IEEE Communication Surveys and Tutorials* and as an associate editor of the *Wiley International Journal of Communication Systems,* and the *Wiley Communication and Mobile Computing*. He served as a cochair of the Signal Processing Track in ISSPIT'14 and served as a cochair of the Transmission Technologies Track in VTC'10-Fall, and the Multimedia and Signal Processing Symposium in CCECE'09. He won the Ontario Graduate Scholarship for Science and Technology in 1997, the Ontario Graduate Scholarship in 1998, 1999, and 2000, and the Communication and Information Technology Ontario (CITO) Graduate Award in 2000. His research interests include radio resource management in wireless networks, multi-hop relaying, cooperative communication, vehicular ad-hoc networks, cognitive radio networks, and wireless sensor networks. His research is sponsored by NSERC, CFI, QNRF, Bell/Aliant, and other governmental and industrial agencies. He is a registered professional engineer (PEng) in the province of Newfoundland, Canada. He is a senior member of the IEEE.

Yuanzhu Chen received the BSc degree from Peking University, in 1999 and the PhD degree from Simon Fraser University, in 2004. He is an associate professor in the Department of Computer Science, Memorial University of Newfoundland, St. John's, Newfoundland. He was deputy head for undergraduate studies in 2012-2015, and deputy head for graduate studies in 2016 to present date. Between 2004 and 2005, he was a postdoctoral researcher with Simon Fraser University. His research interests include computer networking, mobile computing, graph theory, complex networks, Web information retrieval, and evolutionary computation. He is a member of the IEEE.

Octavia A. Dobre (M'05-SM'07) received the engineering diploma and PhD degrees from the Politehnica University of Bucharest (formerly Polytechnic Institute of Bucharest), Romania, in 1991 and 2000, respectively. She was the recipient of a Royal Society Scholarship at Westminster University (2000), and held a Fulbright Fellowship at the Stevens Institute of Technology (2001). Between 2002 and 2005, she was with the Politehnica University of Bucharest and New Jersey Institute of Technology. In 2005, she joined Memorial University, Canada, where she is currently is a full professor and research chair. She was a visiting professor at the Universit de Bretagne Occidentale, France, and Massachusetts Institute of Technology (MIT) (2013). Her research interests include 5G technologies, blind signal identification and parameter estimation techniques, cognitive radio systems, and transceiver optimization algorithms for wireless communications, as well as optical and underwater communications. She has co-authored more than 190 journal and conference papers and gave more than 40 invited talks to industry and academia. She serves as the editor-in-chief of the *IEEE Communications Letters*, as well as an editor of the *IEEE Communications Surveys and Tutorials* and the *IEEE Systems*. She was an editor and a senior editor of the *IEEE Communications Letters*, editor of the *IEEE Transactions on Wireless Communications*, and guest editor for other prestigious journals. She served as general chair of CWIT, and technical co-chair of symposia at numerous conferences, such as IEEE GLOBECOM and ICC. She is the chair of the IEEE ComSoc Signal Processing for Communications and Electronics Technical Committee, the chair of the IEEE ComSoc WICE standing committee, as well as a member-at-large of the Administrative Committee of the IEEE Instrumentation and Measurement Society. She is a senior member of the IEEE.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.