Optimal Stochastic Policies in a network coding capable Ad Hoc Networks

Hayoung Oh
School of Electronic and Engineering, Soongsil University, Seoul 156-743, Korea
[e-mail: hyoh@ssu.ac.kr]
*Corresponding author: Hayoung Oh

Received May 30, 2014; revised April 1, 2014; revised July 18, 2014; accepted October 27, 2014; published December 31, 2014

Abstract

Network coding is a promising technology that increases system throughput by reducing the number of packet transmissions from the source node to the destination node in a saturated traffic scenario. Nevertheless, some packets can suffer from end-to-end delay, because of a queuing delay in an intermediate node waiting for other packets to be encoded with exclusive or (XOR). In this paper, we analyze the delay according to packet arrival rate and propose two network coding schemes, *iXOR (Intelligent XOR)* and *oXOR (Optimal XOR)* with *Markov Decision Process (MDP)*. They reduce the average delay, even under an unsaturated traffic load, through the Holding-χ strategy. In particular, we are interested in the unsaturated network scenario. The unsaturated network is more practical because, in a real wireless network, nodes do not always have packets waiting to be sent. Through analysis and extensive simulations, we show that *iXOR* and *oXOR* are better than the *Distributed Coordination Function (DCF) without XOR (the general forwarding scheme)* and *XOR with DCF* with respect to average delay as well as delivery ratio.

Keywords: Optimal stochastic Policy, Network coding, Markov Decision Process (MDP)

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning(NRF-2014R1A1A1003562).

http://dx.doi.org/10.3837/tiis.2014.12.009

1. Introduction

Linear network coding and exclusive or (XOR) are two major techniques in network coding. Linear network coding transforms the packets with the linear equation in every intermediate node, and the destination node only needs to receive enough of the linear equations in the form of coded packets to successfully decode the original packets [1][2]. It allows the destination node to avoid unnecessarily receiving the same packets several times via the retransmission mechanism after transmission failure. In addition, XOR reduces the number of transmissions because the intermediate node encodes the packets from both sides into one packet and broadcasts it to all original destinations in an Alice-Bob topology or X-topology [3]. As a result, network coding has recently become a spotlight mechanism due to its effect on improving network throughput and proof of the theoretical maximum network capacity. It was first shown under the wired multicasting network [4], and later, the wireless network environment in several papers [5][6][7].

IEEE 802.11 recommended the Distributed Coordination Function (DCF) as the mandatory function for the Media Access Control (MAC) layer of the wireless Local Area Network (LAN)[8]. In wireless communication, transmission failures due to packet collisions can occur when the stations transmit at the same time, because they share the transmission medium—air. To reduce collision probability, DCF uses a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) mechanism, in which the stations count down a random back-off time once during each slot time of the idle channel, and they transmit a packet when the back-off time reaches zero. However, network performance can still decrease due to frequently occurring packet collisions in a saturated traffic scenario. Therefore, most previous network coding schemes have only focused on throughput improvement by reducing the number of transmission packets and collisions in a saturated traffic scenario. However, we are particularly interested in improvement of network coding with DCF in the unsaturated network. The unsaturated network is more practical because, in a real wireless network, nodes do not always have packets waiting to be sent.

Some researchers [9] have investigated throughput performance of physical-layer network coding (PNC) under DCF [10]. They considered a wireless network where two client groups communicate with each other across one relay node, and focused on the unsaturated network. They further derived an approximate closed-form solution for the transmission probability of client nodes that maximizes PNC network throughput. Even though they [9] considered the unsaturated scenario, they derived analytical network throughput for only the PNC under DCF. They did not provide an optimal policy in the relay node to improve performance in the unsaturated wireless network.

Other researchers [11] first reduced the average delay, even under an unsaturated traffic load, with DCF through the Holding-χ strategy concept to maximize the number of network coding chances. However, they only provided a heuristic algorithm for the proper Holding-χ value based on simulation in a specific simple scenario without the optimal solution.

Then [12] proposed a network coding algorithm for a video conference system to minimize the maximal transmission delay during multicast while retaining high throughput at the same time. Zeng et al. [13] proposed enhanced network coding (ENC) considering a delay-energy lower bound on two-way–relay wireless network coding. Dong et al. [14] provided a dynamic network coding model called Dynamic Network Coding with Packet-transmission Delay Guarantee (DNPDG), which effectively controls packet transmission delay in network coding by dynamically determining the coding operation and adjusting the size of data generation.

However, they focused on analyzing the average delay in a specific wireless networks without DCF and did not consider the unsaturated practical scenarios as well as the trade-off between throughput and delay.

Our contribution is average delay reduction as well as delivery ratio improvement using network coding based on IEEE 802.11 distributed coordination function, even in the unsaturated scenario. Therefore, we first analyzed the delay according to packet arrival rate and proposed *Intelligent XOR (iXOR)* using the Holding-χ strategy to get more XOR chances. With the Markov Decision Process (MDP) of optimization theory, we also propose *oXOR (Optimal XOR)* for the optimal Holding-χ value. And we evaluated *oXOR*, *iXOR*, *XOR* and *DCF without XOR (the general forwarding scheme)*. The rest of the paper is organized as follows. In sections 2 and 3, we discuss related work and delay analysis under dynamic traffic scenarios in *DCF without XOR* and *XOR with DCF*. Section 4 presents the heuristic proposed scheme, *iXOR*. We analyze *oXOR* with the Markov Decision Process theory in Section 5. Section 6 evaluates the performance. Finally, Section 7 concludes the paper.

2. Related Work

2.1 DCF (Distributed Coordination Function)

IEEE 802.11 recommends DCF as the standard mechanism to reduce the collision probability of stations that share the transmission medium in wireless networks [8]. In wireless networks, packet collision is unavoidable because the stations are unaware of the time when their competitors transmit, and several stations transmit simultaneously within the propagation delay. Thus, IEEE 802.11 recommends the CSMA/CA mechanism, in which the stations count down a random back-off time once during each slot time of the idle channel, and they transmit when the back-off time reaches 0 to reduce the packet collision probability.

DCF describes two techniques for packet transmission [15]. The default scheme is a two-way handshake technique called the basic access mechanism. This mechanism is characterized by the immediate transmission of a positive acknowledgement (AcK) by the destination station, upon successful reception of a packet transmitted by the sender station. Explicit transmission of an AcK is required since, in the wireless medium, a transmitter can determine if a packet was successfully received by listening for AcK from the destination station.

In addition to basic access, an optional four-way handshake technique, known as the request-to-send/clear-to-send (RTS/CTS) mechanism was standardized. Before transmitting a packet, a station operating in RTS/CTS mode “reserves” the channel by sending a special Request-To-Send short frame. The destination station acknowledges receipt of an RTS frame by sending back a Clear-To-Send frame, after which normal packet transmission and AcK response occurs. Since collision may occur only on the RTS frame, and collision is detected by lack of a CTS response, the RTS/CTS mechanism allows an increase in system performance by reducing the duration of a collision when long messages are transmitted. As an important side effect, the RTS/CTS scheme designed in the 802.11 protocol is suited to combat the problem of so-called Hidden Terminals [16], which occurs when pairs of mobile stations are unable to hear each other. This problem has been specifically considered elsewhere [17][18], where the phenomenon of packet capture was also studied.

2.2 Network Coding

Network coding makes the theoretical maximum network capacity practically achievable by reducing the number of packet transmissions. There are two specific mechanisms in network coding, XOR [3] and linear (random) network coding [1][2]. XOR reduces the number of transmissions in such as way that the intermediate node only broadcasts the packet once after it has been encoded from packets sent by several transmitters, rather than simply forwarding all those packets one by one. Several authors implemented an XOR-bit-level network coding mechanism in a wireless network test-bed, showing that it improves network throughput by 38% [3][8]. More recently, it was shown that analog network coding [5] utilizing signal interference, rather than excluding it, reduces transmission time even more than the traditional bit-level network coding. All these network coding mechanisms could be widely utilized for reliability gain [19][27], multi-hop network gain[20][25], relay network gain [21][24] [34], opportunistic routing [7], peer-to-peer networks [23], efficient content distribution [28][29][33], and energy efficiency [14][22], etc.

In contrast, linear coding [1][2] is where the intermediate nodes forward every packet linear-transformed by a certain linear equation, and destination nodes decode the original packets as long as they receive enough of the linear equations in adequate number. Linear transformation is a multiplication of a vector's so-called coefficient to the bit-pattern of the packet that passes through a station, and is called linear coding when the coefficient becomes 1, whereas it is called random linear coding when the coefficient is less than 1 and larger than 0. (Random) linear coding is generally applied with packet error recovery [7], a multicast scenario, and efficient delivery of urgent messages in Vehicular Ad Hoc Networks and Delay-Tolerant Networking (DTN) [30].

Zeng et al. [13] proposed enhanced network coding with a delay-energy lower bound on a two-way relay wireless network. In ENC, the relay transmits both coded and uncoded packets for reducing delay. Generally, in exchange between two nodes, more energy is consumed to transmit uncoded packets. ENC is a practical algorithm to achieve minimal average delay and a zero packet-loss rate under a given energy constraint. Dong et al. [14] provided a dynamic network coding model called Dynamic Network Coding with Packet-transmission Delay Guarantee (DNPDG). They effectively controls packet transmission delay in network coding by dynamically determining the coding operation and adjusting the size of data generation. And in the coding operation, the model schedules and forwards the packets based on measurement of the current accumulative transmission delay and the service priority of the packet. Lastly, the acknowledgement information feedback with per-hop transmission technique promotes packet transmission of various data generations in the relay nodes. However, these schemes focused on analyzing the fundamental trade-off between average delay and energy/data size and did not consider the unsaturated practical scenario with DCF as well as the trade-off between throughput and delay.

Hui Zhang, et al., proposed the network coding algorithm for a video conference system by minimizing the maximal transmission delay while retaining high throughput at the same time. They first formulate a minimizing delay in multicast scenario with a network coding strategy as an optimization problem [12]. It is based on an algebraic framework represented by a directed graph, G (V, E), with vertex set V representing nodes, and directed edge set E representing links. However, since this paper assumes the multicast scenario is based on the wired Internet, end–to-end delay is mainly caused by the propagation delay along the Internet route and the buffering delay at the intermediate nodes. They did not consider timely throughput and unsaturated scenarios considering DCF.

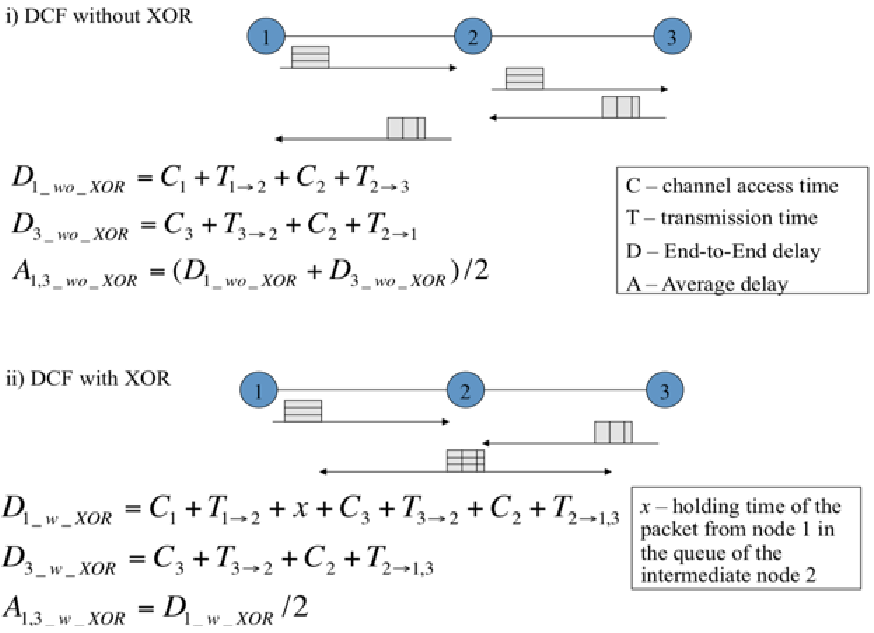
Some researchers have investigated the throughput performance of physical-layer network coding (PNC) under the unsaturated network with DCF [9][10]. They derived an approximate closed-form solution of the transmission probability of client nodes that maximizes throughput. Even though they considered the unsaturation scenario, they only focused on the analytical network throughput results in a specific wireless network scenario where two client groups communicate with each other across one relay node. And they did not provide an optimal policy in the relay node to improve XOR performance in unsaturated wireless networks.

Other researchers [11] first introduced the Holding-χ strategy concept to reduce average delay even under an unsaturated traffic load with DFC by maximizing the number of network coding chances. However, they only provided the proper Holding-χ value heuristically, based on an extensive simulation in a specific scenario without an optimal solution.

3. Delay analysis under dynamic traffic in DCF without XOR and with XOR

Fig. 1 shows the delay comparison between *DCF without XOR (the general forwarding scheme)* and *DCF with XOR*. First of all, i) *DCF without XOR* shows the end-to-end delays taken by each packet generated from source nodes 1 and 3 to destination nodes 3 and 1, $D_{1\text{-wo-XOR}}$ and $D_{3\text{-wo-XOR}}$, respectively. $D_{1\text{-wo-XOR}}$ is composed of the time (C_1) for source node 1 to compete for and grab the channel, the time ($T_{1\rightarrow 2}$) for the packet to be transferred from source node 1 to intermediate node 2, the time (C_2) for intermediate node 2 to compete for and grab the channel, and the time ($T_{2\rightarrow 3}$) for the packet to be transferred from intermediate node 2 to destination node 3. Similarly, $D_{3\text{-wo-XOR}}$ can be derived like $D_{1\text{-wo-XOR}}$. And, $A_{1,3\text{-wo-XOR}}$ means the average delay of the packets from nodes 1 and 3.

In contrast, ii) *DCF with XOR* shows the end-to-end delays of the packets from source nodes 1 and 3 to destination nodes 3 and 1 via broadcast transmission of the coded packet in intermediate node 2: $D_{1\text{-w-XOR}}$, $D_{3\text{-w-XOR}}$, respectively. $D_{1\text{-w-XOR}}$ is composed of the time (C_1) when source node 1 competes for and grabs the channel, the time ($T_{1\rightarrow 2}$) when the packet is transferred from source node 1 to intermediate node 2, the time *(χ)* when the packet from the source node 1 is queued in intermediate node 2 to wait for a packet generated by another source (node 3) to be encoded, the time (C_3) when source node 3 competes for and grabs the channel, the time ($T_{3\rightarrow 2}$) when the packet is transferred from source node 3 to intermediate node 2, the time (C_2) when intermediate node 2 competes for and grabs the channel, and the time ($T_{2\rightarrow 1,3}$) when intermediate node 2 broadcasts and successfully delivers the coded packet to destination nodes 3 and 1. $D_{3\text{-w-XOR}}$ can be also interpreted like $D_{1\text{-w-XOR}}$. However, the packet from node 3 does not wait at intermediate node 2 because the packet from node 1, being coded, is already in the queue of intermediate node 2. And in fact, $D_{3\text{-w-XOR}}$ is not needed because it could be part of $D_{1\text{-w-XOR}}$. As a result, the average delay $A_{1,3\text{-w-XOR}}$ is generally shorter than $A_{1,3\text{-wo-XOR}}$ in the saturated scenario, because XOR reduces the number of transmissions.



$$D_{1\_wo\_XOR} = C_1 + T_{1\rightarrow 2} + C_2 + T_{2\rightarrow 3}$$
$$D_{3\_wo\_XOR} = C_3 + T_{3\rightarrow 2} + C_2 + T_{2\rightarrow 1}$$
$$A_{1,3\_wo\_XOR} = (D_{1\_wo\_XOR} + D_{3\_wo\_XOR})/2$$

$$D_{1\_w\_XOR} = C_1 + T_{1\rightarrow 2} + x + C_3 + T_{3\rightarrow 2} + C_2 + T_{2\rightarrow 1,3}$$
$$D_{3\_w\_XOR} = C_3 + T_{3\rightarrow 2} + C_2 + T_{2\rightarrow 1,3}$$
$$A_{1,3\_w\_XOR} = D_{1\_w\_XOR}/2$$

Fig. 1. E2E delay comparison, i) DCF without XOR and ii) DCF with XOR

However, $A_{1,3\_w\_XOR}$ can be longer than $A_{1,3\_wo\_XOR}$ due to the holding time, χ, according to the packet arrival rate, specially in the unsaturated scenario. Based on this motivation about the relationship between the delay and the packet arrival rate in a network coding–capable wireless network, we propose two strategies, *iXOR* using the heuristic Holding-χ, and *oXOR* with the optimal holding χ value, to get more XOR chances, even in the unsaturated network scenario based on IEEE 802.11 distributed coordination function.

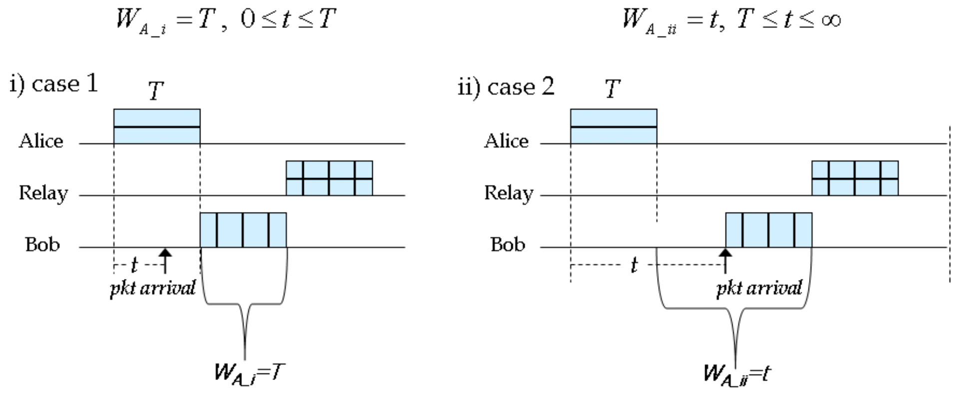
4. iXOR (Intelligent XOR) using Holding-χ strategy in an ad hoc network

To design *iXOR* in an ad hoc network, we define several terms, as follows.

- W_A – the waiting time of Alice's packet in the queue of the intermediate node for Bob's packet to arrive and to be encoded
- W_{A_i} , W_{A_ii} – two cases for waiting time of Alice's packet in the queue of the intermediate node according to the arrival of Bob's packet

Fig. 2. shows the meaning of W_{A_i} and W_{A_ii}. Specifically, i) **Fig. 2(a)** shows when Bob has a packet to transmit at time t, while Alice's packet is transmitted to the relay node. As a result, the waiting time for Alice's packet in the queue of the intermediate node to be encoded with Bob's packet, W_{A_i} , is Bob's packet's transmission time, T. Otherwise, i) **Fig. 2(b)** shows when Bob has a packet to transmit at time t, after Alice's packet is transmitted to the relay node. As a result, the waiting time for Alice's packet (in the queue of the intermediate node) to be encoded with Bob's packet, W_{A_ii}, is the time to Bob's packet generation and transmission from Bob to the relay node, t+T-T. Time t in **Fig. 2(b)** is when Bob has a packet to transmit; the first capital T is the transmission time of Bob's packet from the Bob node to the

intermediate node, and the second capital T is the transmission time of Alice’s packet from the Alice node to the intermediate node. The reason for the negative effect of transmission time T for Alice’s packet is that Bob has to wait until time t to at least have a packet. Therefore, the meaning of the value “t+T-T” is the needed time for Alice’s packet to wait in the middle node to code with Bob’s packet.



(a) case i (b) case ii

Fig. 2. Two cases of waiting time of Alice’s packet at the intermediate relay node, W_{A_i} and W_{A_ii}.

As a result, if we assume that packet arrival rate λ is an exponential distribution, we can calculate the expectation of the waiting time for Alice’s packet $E[W_A]$ by considering $E[W_{A\_i}]$ and $E[W_{A\_ii}]$ in Eq. (1) and Eq. (2).

- $E[W_{A\_i}]$, $E[W_{A\_ii}]$ – the expectation of Alice’s packet waiting time in each case

$$
\begin{aligned}
E\left[W_{A\_i}\right] &= q_{A\_i} \cdot W_{A\_i} = \int_0^T \lambda e^{-\lambda t} \cdot T dt = \left(-e^{-\lambda t} \cdot T\right)\Big|_0^T = \left(e^{-\lambda t} \cdot T\right)\Big|_T^0 = T - e^{-\lambda T} \cdot T \\
E\left[W_{A\_ii}\right] &= q_{A\_ii} \cdot W_{A\_ii} = \int_T^\infty \lambda e^{-\lambda t} \cdot t dt = \left(-\frac{e^{-\lambda t} \cdot (\lambda t + 1)}{\lambda}\right)\Bigg|_T^\infty = \frac{e^{-\lambda T} \cdot (\lambda T + 1)}{\lambda}
\end{aligned} \tag{1}
$$

- $E[W_A]$ – the expectation of Alice’s packet waiting time

$$
E\left[W_A\right] = E\left[W_{A\_i}\right] + E\left[W_{A\_ii}\right] = \int_0^T \lambda e^{-\lambda t} \cdot T dt + \int_T^\infty \lambda e^{-\lambda t} \cdot t dt = T + \frac{e^{-\lambda T}}{\lambda} \tag{2}
$$

Each probability of case i) and case ii), q_{A_i} and q_{A_ii}, is a cumulative distribution probability according to each duration of the probability density function of the exponential distribution, $\lambda e^{-\lambda t}$ in Eq. (3). T is the channel occupation time (DIFS+backoff+Tx+SIFS+AcK+round-trip propagation delay), and t is the generation time for Bob’s packet.

$$
\begin{aligned}
q_{A\_i} &= \int_0^T \lambda e^{-\lambda t} dt = 1 - e^{-\lambda T} \\
q_{A\_ii} &= \int_T^\infty \lambda e^{-\lambda t} dt = e^{-\lambda T}
\end{aligned} \tag{3}
$$

Along the same lines, we can also define the waiting time for Bob's packet in the queue of the Bob node as follows. Fig. 3. shows the meanings for W_{B_i} and W_{B_ii}.

- W_B - waiting time for Bob's packet in the queue of the Bob node for Alice's packet to be delivered to the intermediate node
- W_{B_i}, W_{B_ii} – two cases of waiting time for Bob's packet in the queue of the Bob node according to Bob's packet arrival

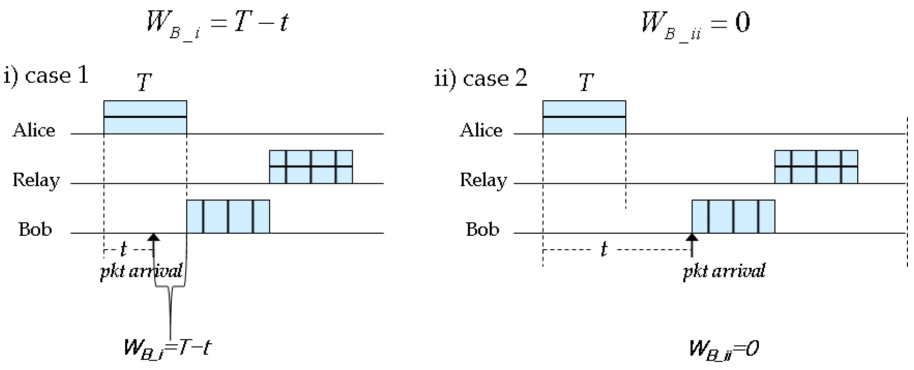


Fig. 3. Two cases of waiting time of Bob's packet in the Bob node, W_{B_i} and W_{B_ii}.

As a result, we can also calculate the expectation of Bob's packet's waiting time $E[W_B]$ considering $E[W_{B\text{-}i}]$ and $E[W_{B\text{-}ii}]$ in Eq. (4).

- $E[W_{B\text{-}i}]$, $E[W_{B\text{-}ii}]$ – the expectation of Bob's packet's waiting time in each case
- $E[W_B]$ – the expectation of Bob's packet's waiting time

$$
\begin{aligned}
E[W_B] &= E[W_{B\_i}] + E[W_{B\_ii}] = q_{B\_i} \cdot W_{B\_i} + q_{B\_ii} \cdot 0 = E[W_{B\_i}] \\
&= \int_0^T \lambda e^{-\lambda t}(T - t)dt = \left.\frac{e^{-\lambda t}(\lambda(t - T) + 1)}{\lambda}\right|_0^T = T + \frac{e^{-\lambda T} - 1}{\lambda}
\end{aligned}
\tag{4}
$$

Finally, we can get the total waiting time of *DCF with XOR* in Eq.(5) since in this paper, the overall waiting time is the summation of each holding packet for the network coding in any queues due to the busy medium based on IEEE 802.11 distributed coordination function.

$$
E[W_{XOR}] = E[W_A] + E[W_B] = 2T + \frac{2e^{-\lambda T} - 1}{\lambda}
\tag{5}
$$

Fig. 4. shows the comparisons of $E[W_A]$ and $E[W_B]$ for each packet according to the packet arrival rate, λ, and packet size T. And Fig. 5. shows $E[W_{XOR}]$ of the XOR exchange system. It shows that the network coding introduces a longer delay when packet arrival rate λ is very low. Therefore, we need to find the proper Holding-χ at the intermediate node to reduce the average delay.

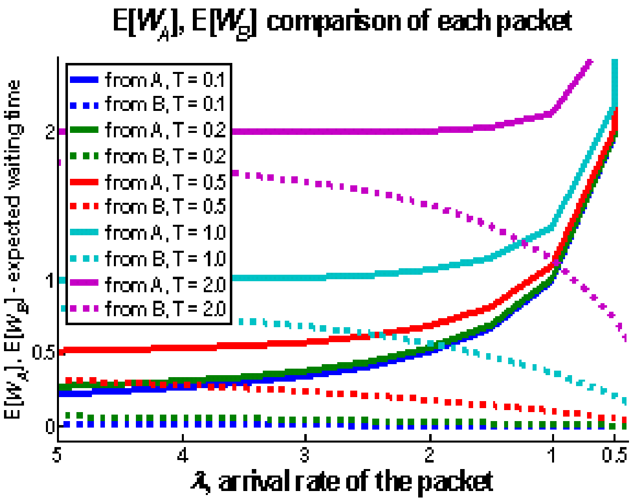


Fig. 4. $E[W_A]$ and $E[W_B]$ comparisons of each packet

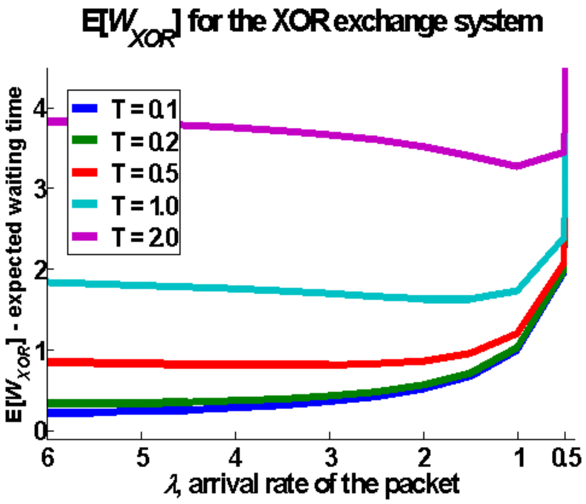


Fig. 5. $E[W_{XOR}]$ for the XOR exchange system

To compare *DCF without XOR* and *DCF with XOR*, we in detail explain the case studies of *DCF without XOR*. These are classified into *DCF without RTS/CTS* and *DCF with RTS/CTS* as shown in **Figs. 6** and **7**, respectively.

1) *DCF without RTS/CTS*

This is classified into three cases according to Bob's packet arrival rate. In the saturation and middle-saturation scenarios with $0 \le t \le T$ and $T \le t \le 2T$, respectively, we again have two cases based on the channel taking chances between the Alice packet and Bob packet. In case of the unsaturated scenario with $2T < t$, there is a single case where each packet from Alice and Bob is transmitted to each destination in the order named.

And we can calculate the expected total waiting time based on the *DCF without RTS/CTS* of **Fig. 6** in Eq. (6). The probability σ is so low that the second case of the middle-saturation scenario can be ignored. Therefore, only the two cases of the saturation and the first case of middle-saturation scenarios are considered to get the expectation delay of $E[W_{DCF}]$.

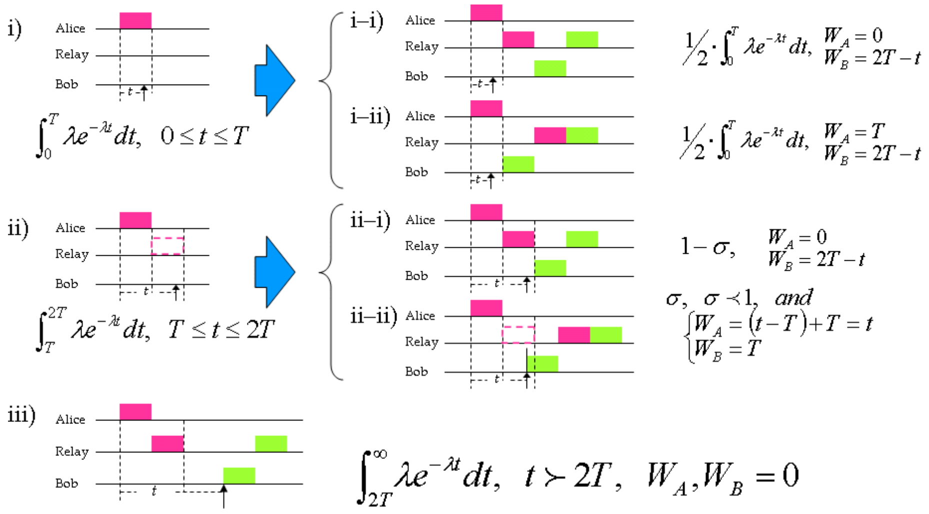


Fig. 6. The case studies of *DCF without XOR* (w/o RTS and CTS)

$$
\begin{aligned}
E[W_{DCF}] &= E[W_A] + E[W_B] \\
&\approx \left(\int_0^T \lambda e^{-\lambda t} \, \tfrac{1}{2} T dt + \int_T^{2T} \lambda e^{-\lambda t} \sigma t dt \right) + \\
&\left(\int_0^T \lambda e^{-\lambda t} (2T - t) dt + \int_T^{2T} \lambda e^{-\lambda t} (1 - \sigma)(2T - t) dt + \int_T^{2T} \lambda e^{-\lambda t} \sigma T dt \right) \\
&\approx \int_0^T \lambda e^{-\lambda t} \, \tfrac{1}{2} T dt + \int_0^{2T} \lambda e^{-\lambda t} (2T - t) dt \qquad (\sigma \prec 1) \\
&= -\tfrac{1}{2} \cdot e^{-\lambda t} T \Big|_0^T + \left. \frac{e^{-\lambda t}(1 + \lambda(t - 2T))}{\lambda} \right|_0^{2T} \\
&= \tfrac{T}{2}\left(5 - e^{-\lambda T}\right) + \tfrac{1}{\lambda}\left(e^{-2\lambda T} - 1\right)
\end{aligned} \tag{6}
$$

2) DCF with RTS/CTS

With the *DCF with RTS/CTS* exchange protocol, we only think about i-ii), ii-ii) and iii) cases of **Fig. 6**, as shown in **Fig. 7**. The relay node can calculate the transmission duration of the Alice packet after receiving the RTS control packet from the Alice node. And the relay node can send CTS with information about the next transmission turn and time, as well as acknowledge the RTS. For example, in ii) and ii) cases of **Fig. 7**, Bob (receiving the CTS from the relay node) can transmit the corresponding packet to the relay node at the given time after CTS. However, if the corresponding packet from Bob is lately generated under the severe unsaturated scenario, as shown in case iii) of **Fig. 7**, the relay node should just forward the Alice packet to the Bob node to minimize the average end-to-end delay. Eq.(7) is the expected total waiting time based on the *DCF with RTS and CTS* of **Fig. 7**.

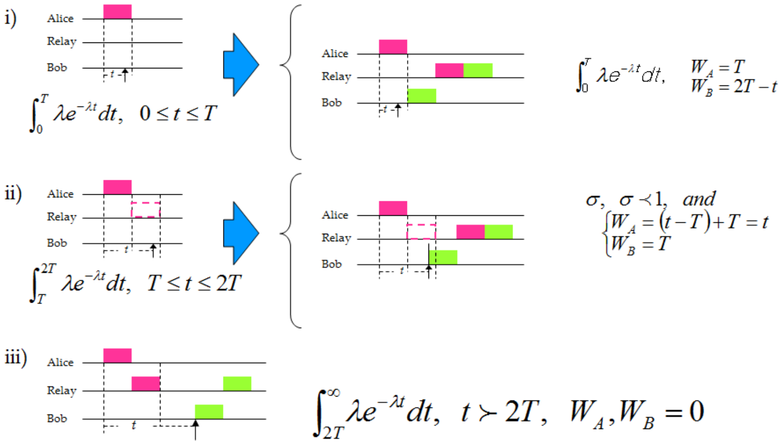


Fig. 7. The case studies of DCF (w/ RTS and CTS)

$$
\begin{aligned}
E\left[W_{DCF\_RTS/CTS}\right] &= E\left[W_A\right] + E\left[W_B\right] \\
&\approx \left(\int_0^T \lambda e^{-\lambda t} T dt + \int_T^{2T} \lambda e^{-\lambda t} \sigma t dt\right) + \\
&\left(\int_0^T \lambda e^{-\lambda t}(2T - t)dt + \int_T^{2T} \lambda e^{-\lambda t} \sigma T dt\right) \\
&\approx \int_0^T \lambda e^{-\lambda t}(3T - t)dt + \int_0^{2T} \lambda e^{-\lambda t}(T + t)\sigma dt \qquad (\sigma \prec 1)
\end{aligned} \tag{7}
$$

At last, we can get the cross point of the light blue, which gives the motivation for the proposed scheme, *iXOR*, as shown in **Fig. 8**. *iXOR* opportunistically exchanges the function from *XOR* to *DCF*, or conversely, according to packet arrival rate λ. Specifically, in *iXOR*, the relay node holds Alice's packet until it gets Bob's packet, as long as the arrival rate of the packet is larger than the cross point of the light blue. On the other hand, the relay node does not hold Alice's packet but forwards it directly to the destination based on *DCF* when the arrival rate of the packet is smaller than the cross point of the light blue to reduce the average delay.

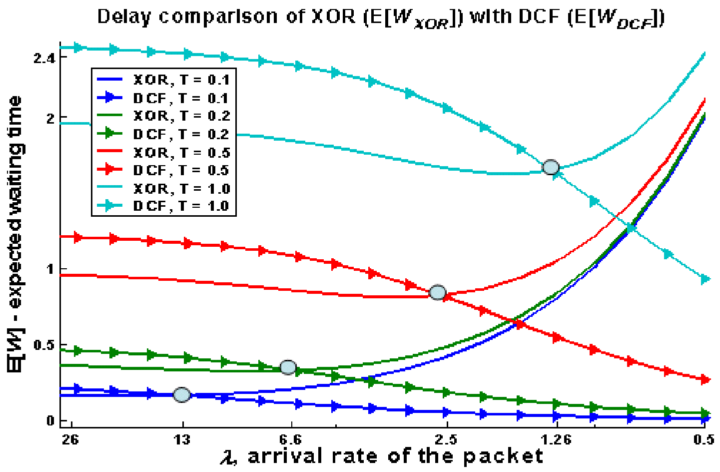


Fig. 8. Delay comparison between
DCF with XOR (**E[W_{XOR}]**) and *DCF without XOR* (**E[W_{DCF}]**)

Therefore, motivated by the previous part and **Fig. 8**, we focus on the holding time, χ, to reduce the average delay with *iXOR* in the various scenarios, according to packet arrival rate λ. **Fig. 9** shows *iXOR* using the Holding-χ strategy in ad hoc networks. The proposed scheme considers three scenarios according to packet arrival rate λ. In the first scenario with the low enough arrival rate λ, after a packet from node 1 has arrived at intermediate node 2, intermediate node 2 cannot XOR-encode with another packet from node 3. Even though intermediate node 2 waits for the Holding-χ for another packet from node 3, it is not delivered to intermediate node 2 within the Holding-χ. Specifically, time t for packet arrival from node 3 is longer than the Holding-χ ($t>\chi$ *and* $\chi\neq0$). In the second scenario, with the medium arrival rate, intermediate node 2 can XOR-encode the two packets from nodes 1 and 3 because, after a packet from node 1 arrives at intermediate node 2, the packet from node 3 arrives at intermediate node 2 within Holding-χ ($t<\chi$ *and* $\chi\neq0$). And in the third scenario with the high enough arrival rate, intermediate node 2 can instantly XOR-encode the packets from nodes 1 and 3 regardless of Holding-χ because, after a packet from node 1 arrives at intermediate node 2, the packet from node 3 immediately arrives at intermediate node 2 before Holding-χ ($\chi=0$).

In the proposed scheme the intermediate node opportunistically uses the Holding-χ strategy according to the packet arrival rate to get more XOR chances. Heuristically, we can get the proper holding time, χ=0.03 ms, through the extensive simulations.

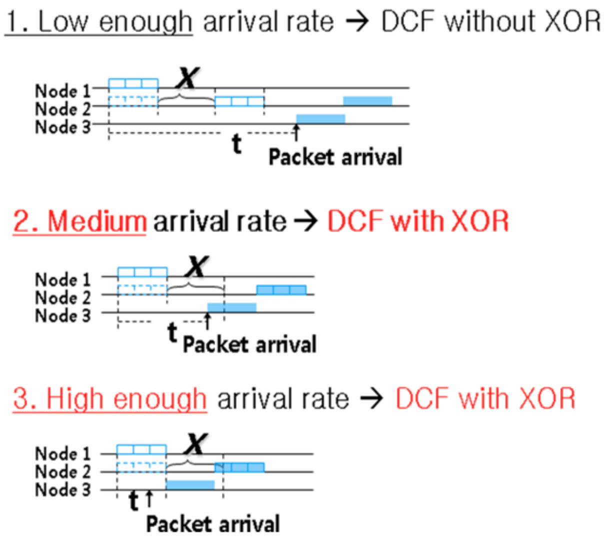


Fig. 9. iXOR using Holding-χ Strategy

5. oXOR (Optimal XOR) using Markov Decision Process

To compare the difference between the previous heuristic algorithm, *iXOR*, and an optimal solution, we develop optimal policies that yield a transmit/wait decision at each time instant in the relay node to optimize the timely throughput over DCF. We define our objective as the maximum long-run average throughput and minimum delay on a per-unit-time basis. Thus, we proposed *oXOR* based on the optimal policies. In this scheme, the relay node can transmit and incur a transmission cost for forwarding a single packet. It can also wait in the hope that a codable packet will arrive, which allows the transmission cost for broadcasting the XOR-ed packet as well as the latency cost for holding a packet. For that, in **Fig. 10** the relay node maintains two queues q_A and q_B, such that q_A and q_B store packets that are needed to be delivered to the nodes *Alice* and, *Bob* respectively. If both the queues are not empty, then, it can forward the two packets from these queues by performing a bitwise XOR operation. However, if one of the queues has packets to transmit and the second queue is empty, the relay node should wait the optimal time for a coding opportunity.

To compare *oXOR* with *iXOR*, we first consider Alice and Bob topology with a single relay node, and assume that arrivals into both the queues follow independent Bernoulli processes [26]. We find that the optimal policy is a stationary queue-length threshold policy with one threshold for each queue at the relay. Its action is simple: if a coding opportunity exists, code and transmit; else if the threshold for that queue is reached, it transmits a packet. We show how to find the optimal thresholds, and find the exact expressions for the expected throughput and latency based on the stationary distribution of the Markov Chain when controlled by this policy.

5.1 Markovian Model

To develop a strategy for the relay to decide at every transmission opportunity, for its best course of action, we use a Markov Decision Process (MDP) model. For $n = A, B$ and $t = 0, 1, 2, \ldots$, let Q_t^n be the number of packets in a queue of n at the end of the time slot t, just before an opportunity to transmit as showin in Fig.10.

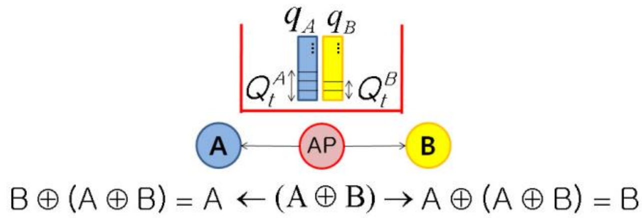


Fig. 10. Wireless network coding in a downlink scenario

Let A_t be the action chosen at the end of the *t*th time slot with $A_t = 0$ implying that the action is to do nothing but wait, and $A_t = 1$ implying that the action is to transmit. We define costs for the transmission and latency. Let C_{tx_XOR} be the transmission cost for broadcastingthe XOR-ed packet. Let C_{tx_fwd} be the transmission cost for forwarding a single packet and let C_w be the latency cost for holding a packet for a length of time which is equal to one slot. They are calculated by the expected transmission count (ETX) and the expected transmission time (ETT) [31][32] metrics. C_{tx_XOR} is the ETT for the relay node to broadcast a XOR-ed packet with a low rate to the two destinationnodes while C_{tx_fwd} is the ETT for the relay node to unicast a single packet to one destination node. Without loss of generality, we assume that if a packet was transmitted in the same slot through which it arrived, its latency cost is zero. On the other hand, if a packet was transmitted through a different slot from which it arrived, C_w is ETT/ETX. Our objective is to derive an optimal policy that maximizes the long-run average throughput while minimizing the average delay per slot. For this, we define the MDP $\{(Q_t, A_t), t \geq 0\}$ where $Q_t =$

(Q_t^A, Q_t^B) is the state of the system and A_t is the control action chosen at a time t. The state space (i.e. all possible values of Q_t) is the set{(i, j - integers): i≥0, 0≤j≤1 or j≥0, 0≤i≤1} as shown in **Fig. 11**. Let $C(Q_t, A_t)$ be the cost incurred at time t in Eq. (8) if the action A_t is taken when the system is in state Q_t.

$$C(Q_t, A_t) = c_w([Q_t^A - A_t]^+ + [Q_t^B - A_t]^+) + c_{tx}A_t\,,$$
$$\left\{\begin{array}{ll} c_{tx} = c_{tx\_XOR}\,, & \text{if XOR-ing} \\ c_{tx} = c_{tx\_fwd}\,, & \text{if forwarding} \end{array}\right\} \tag{8}$$

where, $[x]^+ = \max(x,0)$. For the MDP $\{(Q_t, A_t), t \geq 0\}$, the transition probability (P_1, P_2, P_3) from a state Q_t to Q_{t+1} that is associated with the action $A_t \in \{0,1\}$. This can be derived from the different Bernoulli arrival rates (p_A, p_B) and the optimal policy is of the threshold type as in shown **Fig. 11**.

There exist the optimal thresholds i_A^* and j_B^* of each queue size for the packets to wait. The optimal deterministic action in states (i, 0) is to wait if $i \leq i_A^*$ and to transmit without coding if $i > i_A^*$. While in state (0, j), it is to wait if $j \leq j_B^*$, and to transmit without coding if $j > j_B^*$. Therefore, the optimal thresholds i_A^* and j_B^* for the long-run average throughput per slot are $(i_A^*, j_B^*) =$

$$\arg\max_{i_A, j_B} \frac{2\cdot\mu(i_A, j_B) + \sigma(i_A, j_B)}{c_{tx\_XOR}\,\mu(i_A, j_B) + c_{tx\_fwd}\,\sigma(i_A, j_B) + c_w\,\rho(i_A, j_B)} \tag{9}$$

Eq. (9) shows the way how to derive an optimal policy that maximizes the long-run average throughput while minimizing the average delay per slot. To maximize the average throughput, the denominator with the average delay should be reduced.

As shown in **Fig. 12**, the expected number of XOR-ed transmissions per slot is the states with the red circles. It can be derived by,

$$\mu(i_A, j_B) = p_B \sum_{i=1}^{i_A} \pi_{i,0} + p_A \sum_{j=1}^{j_B} \pi_{0,j} \tag{10}$$

The expected number of forwarding transmissions per slot is the states with the yellow circles. It can be derived by,

$$\sigma(i_A, j_B) = p_A(1-p_B)\pi_{i_A,0} + p_B(1-p_A)\pi_{0,j_B} \tag{11}$$

The average number of packets in the system at the beginning of each slot is the states with the blue circles. It can be derived by,

$$\rho(i_A, j_B) = \sum_{i=1}^{i_A} \pi_{i,0} + \sum_{j=1}^{j_B} \pi_{0,j} \tag{12}$$

To complement the optimal solution with the steady-state probabilities of the Markov chain from the MDP problem, we define two variables. Let X_t^A be the number of packets from node A at the beginning of the tth slot before any arrival or transmission. It is crucial to note that, this observation time is different from the time when the MDP is observed. Then, the bivariate stochastic process $\{(X_t^A, X_t^B), t \geq 0\}$ is a discrete-time Markov chain only consisting of the states with the white and blue circles as shown in **Fig. 12**. The states are (0, 0), (1, 0),(2, 0), . . ., (i_A , 0), (0, 1), (0, 2), . . ., (0, i_B). Define α as a parameter such that, $\alpha = (1-p_B)p_A/(1-p_A)p_B$.

And let $\pi_{i,j}$ be the steady-state probabilities of the Markov chain. To prove the balance equations for $0 < i \le i_A$ and $0 < j \le j_B$, we define two equations as follows:

$$\pi_{i,0} = \alpha^i \pi_{0,0} \tag{13}$$

$$\pi_{0,j} = \pi_{0,0} / \alpha^j \tag{14}$$

Since, $\pi_{0,0} + \sum_{i,j} \pi_{i,0} + \pi_{0,j} = 1,$ we have,

$$\pi_{0,0} = 1/\{(\frac{1-\alpha^{i_A+1}}{1-\alpha}) + (\frac{1-1/\alpha^{j_B+1}}{1-\frac{1}{\alpha}}) - 1\} \tag{15}$$

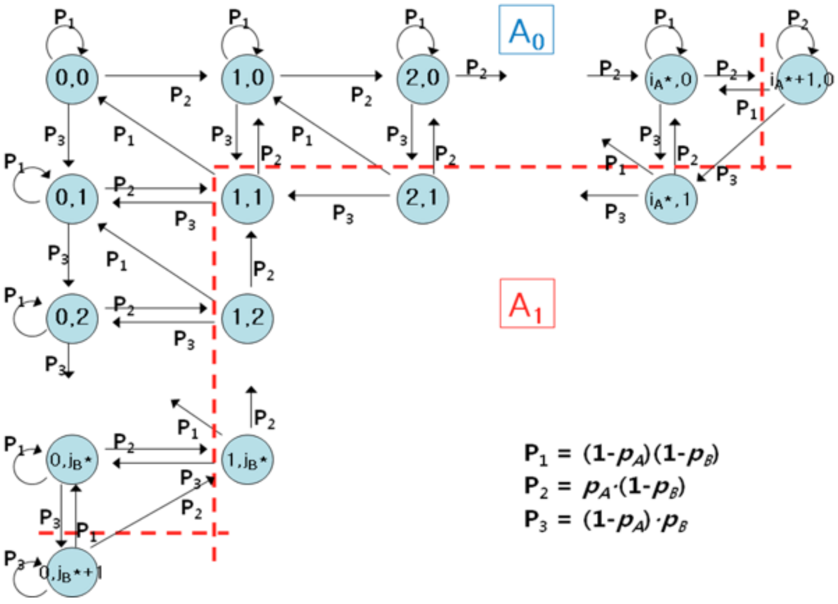


Fig. 11. Markov Decision Process Model for *oXOR*

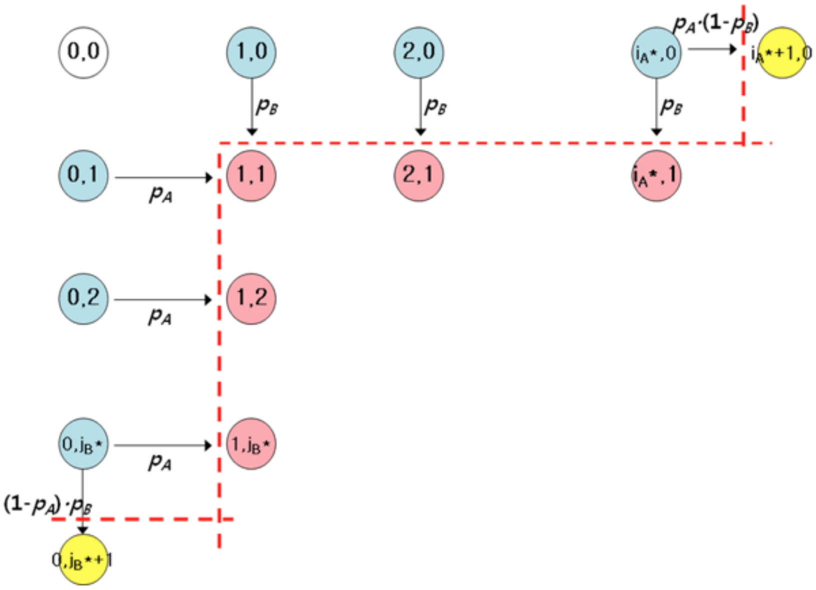


Fig. 12. A steady-sate transition diagram for *oXOR* with the optimal thresholds

6. Performance evaluation

To get the proper holding deadline, χ, we evaluate the average delay and the delivery ratio of *DCF without XOR*, *DCF with XOR*, *iXOR* and *oXOR* with the event-driven simulator NS-3 under the various scenarios. The parameters used in simulations are summarized in **Table 1**.

Table 1. Simulation Properties

Meaning	**Value**
Number of nodes	100
Packet arrival rate	Poisson arrival
Transmission time	12msec
Mean inter-arrival time	48msec
Packet size	1500byte
Data rate	1Mbps
Fading model	Rayleigh fading signal
Holding-χ	0.3msec

Fig. 13, **14** and **15** show the average delay of *DCF*, *XOR*, *iXOR*$_1$ *(w/o RTS/CTS)*, *iXOR*$_2$ *(w RTS/CTS)* and *oXOR* according to the holding χ, "chi", in unicast, multicast and broadcast scenarios, respectively. Because *DCF* and *XOR* do not involve "chi" parameter, the average delays of them are unchanged according to the "chi". And when the "chi" is zero, *iXOR*$_1$, *iXOR*$_2$ and *oXOR* work exactly the same as *XOR*. While the "chi" is increased until 0.3msec, the average delay of *iXOR* and *oXOR* schemes outperforms *XOR*. However, if the "chi" value is longer than 0.3msec, the holding-χ can be rather overhead in the unsaturated scenario. The reason why *oXOR* is better than *iXOR*$_1$ and *iXOR*$_2$, it can find the optimal holding time based on the MDP process. However, the gain between *oXOR* and *iXOR* schemes is not quite

amazing since the holding-χ (i.e., χ =0.3msec) based on the extensive simulations of the previous our work [11] can reach the optimal value. Therefore, the contribution of this paper is the average delay reduction as well as delivery ratio improvement using the Markov Decision Process (MDP) of optimization theory firstly in the unsaturated scenario. And through the evaluation results, our previous heuristic algorithm (i.e., *iXOR* [11]) is proved meaningful since it can reach the optimal solution (i.e., *oXOR*) of this paper with the small gap.

Fig. 16 shows the average delay according to the packet arrival rate λ, "lambda" in unicast scenario. It shows that *iXOR* and *oXOR* schemes outperform *XOR* regardless of the "lambda". **Fig. 17** show the busy ratio of *DCF*, *XOR*, $iXOR_1$ *(w/o RTS/CTS)*, $iXOR_2$ *(w RTS/CTS)* and *oXOR* according to the holding χ, "chi", in unicast. Busy ratio means the wireless channel occupation ratio. *iXOR* and *oXOR* schemes outperform *DCF* and *XOR* because they can reduce the number of transmissions through more coding chances. Due to the similar reasons, the delivery ratios of *iXOR* and *oXOR* schemes are also better than *DCF* and *XOR* as shown **Fig. 18**. **Fig. 19** shows the delivery ratio according to the number of nodes to consider multiuser effects. Apparently, there is a trade-off between throughput and collisions in multiuser case. More users will lead to higher throughput gain whereas it will bring much more collisions. However, generally multiuser increase network coding chances in *iXOR* and *oXOR* schemes. Specifically, $iXOR_2$ is very similar with *oXOR* since it can solve the collision problems with RTS and CTS than $iXOR_1$.

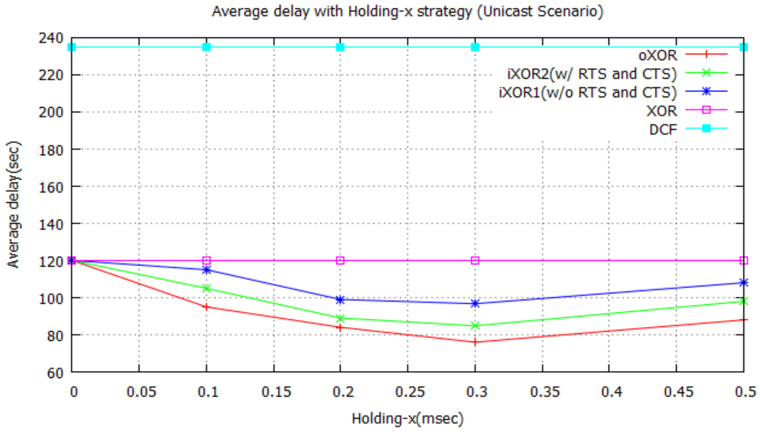


Fig. 13. Average delay with holding-χ in unicast scenario

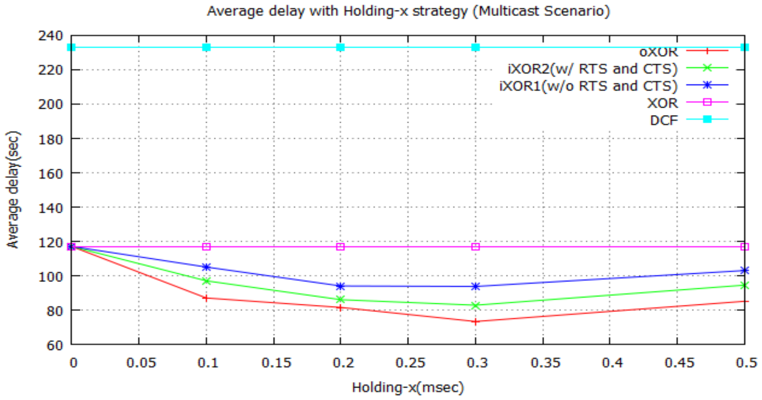


Fig. 14. Average delay with holding-χ in multicast scenario

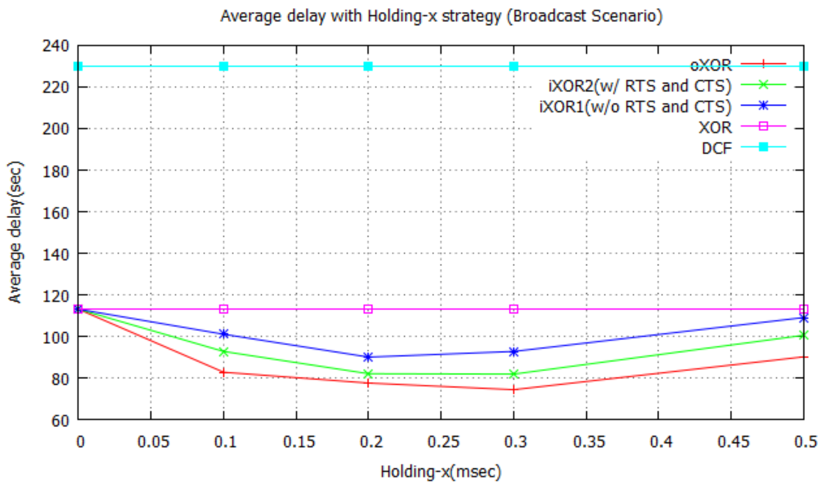


Fig. 15. Average delay with holding-χ in broadcast scenario

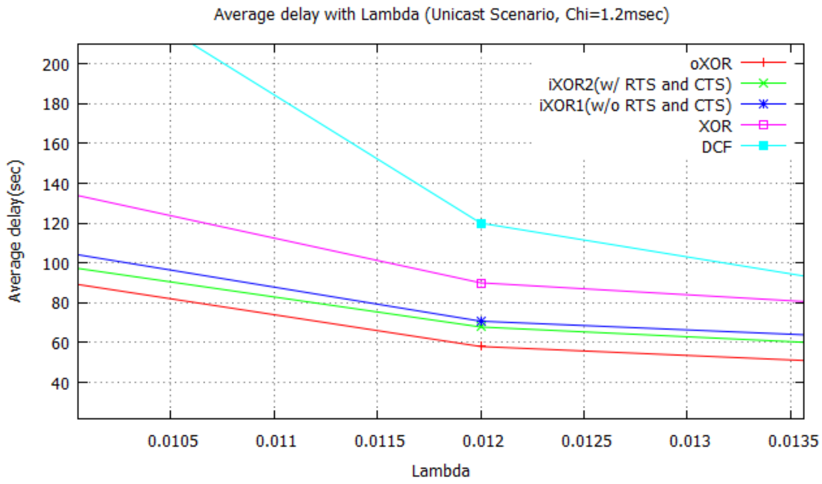


Fig. 16. Average delay according to lambda (=packet arrival rateλ)

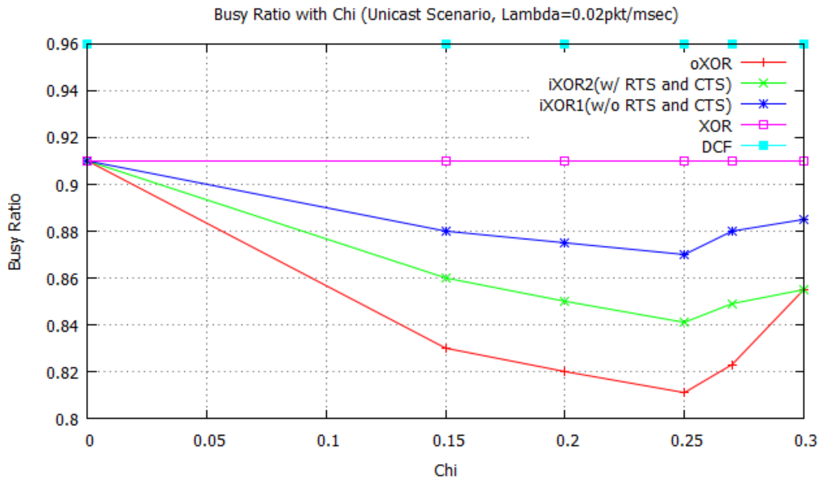


Fig. 17. Busy ratio according to chi (=holding-χ)

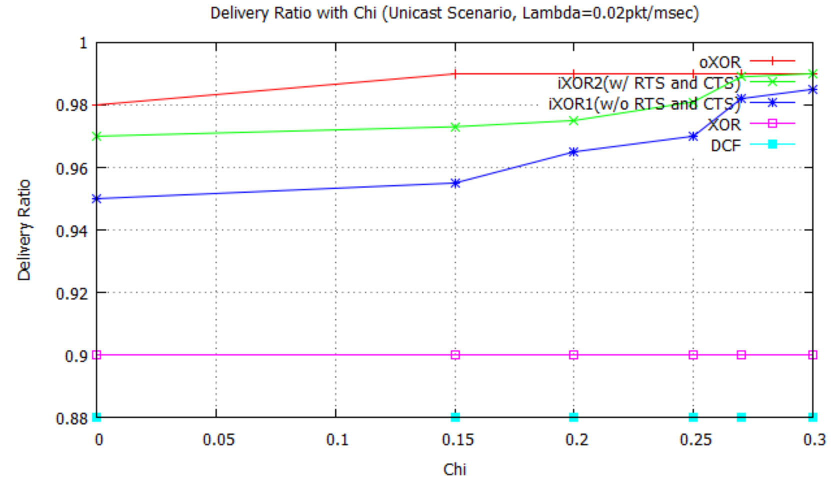


Fig. 18. Delivery ratio according to chi (=holding-χ)

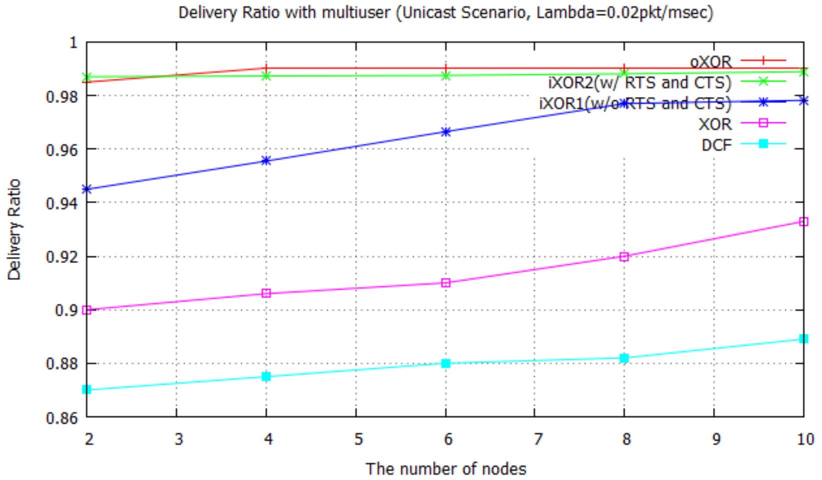


Fig. 19. Delivery ratio according to the number of nodes

Conclusion

In this paper, we analyzed the average delay and timely throughput according to the packet arrival rate λ in a network coding capable wireless network with Markov Decision Process (MDP) and proposed the *iXOR* and *oXOR* using holding-χ strategy in ad-hoc network. Through the simulation and analysis with MDP, we can get the optimal holding-χ value. And we found *iXOR* and *oXOR* outperform *DCF* and *XOR* schemes if the intermediate node opportunistically XOR-encodes the packets with the optimal holding-χ.

Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning(NRF-2014R1A1A1003562).

References

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

[2] Christina Fragouli, JorgWidmer, and Jean-Yves Le Boudec, "Efficient Broadcasting Using Network Coding," *IEEE/ACM TRANSACTIONS ON NETWORKING*, VOL. 16, NO. 2, APRIL 2008. Article (CrossRef Link)

[3] SachinKatti, Hariharan Rahul, Wenjun Hu, Dina Katabi and et al, "XORs in the Air: Practical Wireless Network Coding," *ACM SIGCOMM 2006*, Pisa, Italy, September 2006. Article (CrossRef Link)

[4] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network Information Flow," *IEEE Transactions on Information Theory*, 2000. Article (CrossRef Link)

[5] SachinKatti, ShyamnathGollakota, and Dina Katabi, "Embracing Wireless Interference: Analog Network Coding," *ACM SIGCOMM*, 2007. Article (CrossRef Link)

[6] SachinKatti, Dina Katabi, Wenjun Hu, Hariharan Rahul and Muriel Medard,"Practical Network Coding for WirelessEnvironments," in *Proc. of Allerton Conference on Communication, Control, and Computing*, September 2005. Article (CrossRef Link)

[7] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading Structure for Randomness in Wireless Opportunistic Routing," in *Proc. of ACM SIGCOMM*, 2007. Article (CrossRef Link)

[8] IEEE, "International Standard for Information Technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific Requirements- Part 11: Wireless LAN Medium." Article (CrossRef Link)

[9] Lin Shijun, Xiamen FujianandFu Liqun, "Unsaturated Throughput Analysis of Physical-Layer Network Coding Based on IEEE 802.11 Distributed Coordination Function," *IEEE Transactions on Wireless Communications*, 2013. Article (CrossRef Link)

[10] Emad and E. Eylem, "Single hop IEEE 802.11 DCF analysis revisited: accurate modeling of channel access delay and throughput for saturated and unsaturated traffic cases," *IEEE Trans. Wireless Commun.*, vol. 10, no. 10, pp. 3256–3266, 2011. Article (CrossRef Link)

[11] Hayoung Oh, Junjie Lee, Suchul Lee, and Chong-kwon Kim, "iXOR : Intelligent XOR using Holding-x Strategy in Ad hoc Networks," *IEEE VTC-Spring*, May 2011. Article (CrossRef Link)

[12] Hui Zhang, Jin Zhoub, Zhen Chenb, and Jun Lib, "Minimizing Delay for Video Conference with Network Coding," *ACM Sigcomm*, 2009. Article (CrossRef Link)

[13] Zeng, Hongyi, and Wei Chen. "Delay-Energy lower bound on Two-Way Relay Wireless Network Coding," *arXiv preprint arXiv:1401.6275*, 2014. Article (CrossRef Link)

[14] DONG, Zanqiang, Subin SHEN, and L. I. Yangqun, "Dynamic Network Coding with Delay-guarantee in Multi-hop Wireless Networks," *Journal of Computational Information Systems 9.4*, 1529-1538, 2013. Article (CrossRef Link)

[15] G. Bianchi, "Performance Analysis of The IEEE 802.11 Distributed Coordination Function," *IEEE J. Selected Areas in Comm.*, vol. 18, no. 3, pp. 535-547, 2000. Article (CrossRef Link)

[16] F. Cali, M. Conti, and E. Gregori, "IEEE 802.11 Wireless LAN: Capacity Analysis and Protocol Enhancement," in *Proc. of IEEE INFOCOM '98 Conf.*, pp. 142-149, 1998. Article (CrossRef Link)

[17] F. Cali, M. Conti, and E. Gregori, "IEEE 802.11 Protocol: Design and Performance Evaluation of

an Adaptive Backoff Mechanism," *IEEE J. Selected Areas in Comm.*, vol. 18, no. 9, pp. 1774-1786, 2000. Article (CrossRef Link)

[18] P. Chatzimisios, A.C. Boucouvalas, and V. Vitsas, "Performance Analysis of IEEE 802.11 DCF in Presence of Transmission Errors," in *Proc. of IEEE Int'l Conf. Comm.*, pp. 3854-3858, 2004. Article (CrossRef Link)

[19] MajidGhaderi, Don Towsley, Jim Kurose, "Reliability Gain of Network Coding in Lossy Wireless Networks," *IEEE Infocom 2008*. Article (CrossRef Link)

[20] DimitriosKoutsonikolas, Y. Charlie Hu, Chih-Chun Wang, "An Empirical Study of Performance Benefits of Network Coding in Multihop Wireless Networks," *IEEE* Infocom 2009. Article (CrossRef Link)

[21] Jin Zhang, Qian Zhang, "Cooperative Network Coding-Aware Routing for Multi-Rate Wireless Networks," *IEEE Infocom 2009*. Article (CrossRef Link)

[22] C. Fragouli, J. Widmer, and J.-Y. L. Boudec, "A network coding approach to energy efficient broadcasting: From theory to practice," *IEEE INFOCOM*, Barcelona, Spain, Apr. 2006. Article (CrossRef Link)

[23] Xinyu Zhang, Baochun Li, "On the Market Power of Network Coding in P2P Content Distribution Systems," *IEEE Infocom 2009*. Article (CrossRef Link)

[24] Y. E. Sagduyu and A. Ephremides, "On Joint MAC and Network Coding in Wireless Ad Hoc Networks," *IEEE Transactions on Information Theory*, 2007. Article (CrossRef Link)

[25] Jin Jin, Baochun Li, Taegon Kong, "Is Random Network Coding Helpful in WiMAX?," *IEEE Infocom 2008*. Article (CrossRef Link)

[26] Yu-Pin Hsu et al., "Opportunities for Network Coding: To Wait or Not to Wait," *2011 IEEE International Symposium on Information Theory Proceedings*, 2011. Article (CrossRef Link)

[27] Jilin Le, John C.S. Lui, Fellow, IEEE, and Dah-Ming Chiu, "On the Performance Bounds of Practical Wireless Network Coding," *IEEE TRANSACTIONS ON MOBILE COMPUTING*, VOL. 9, NO. 8, AUGUST 2010. Article (CrossRef Link)

[28] C. Gkantsidis and P. Rodriguez, "Network coding for large scale content distribution," *IEEE INFOCOM*, Miami, FL, Mar. 2005. Article (CrossRef Link)

[29] Uichin Lee, JoonSang Park, Joseph Yeh, Giovanni Pau, Mario Gerla, "CodeTorrent: Content Distribution using Network Coding in VANET," *MobiShare'2006*. Article (CrossRef Link)

[30] Joon-Sang Park, Uichin Lee, Soon Young Oh, Mario Gerla and Desmond Lun, "Emergency Related Video Streaming in VANETs using Network Coding," *UCLA CSD TECHNICAL REPORT: TR-070016*. Article (CrossRef Link)

[31] Douglas S. J. De Couto, et al, "A High-Throughput Path Metric for Multi-Hop Wireless Routing," in *Proc. of ACM Mobicom*, Sep.2003. Article (CrossRef Link)

[32] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. of ACM Mobicom*, Sep.2004. Article (CrossRef Link)

[33] I-Hong Hou, Yu-En Tsai, TarekAbdelzaher, Indranil Gupta, "AdapCode: Adaptive Network Coding for Code Updates in Wireless Sensor Networks," *IEEE Infocom 2008*. Article (CrossRef Link)

[34] Tao Cui, Lijun Chen, Tracey Ho, "Energy Efficient Opportunistic Network Coding for Wireless Networks," *IEEE Infocom 2008*. Article (CrossRef Link)

Hayoung Oh received the B.S. degree in Computer Science from Duksung Womans University and the M.S. degree in the School of Computer Science and Engineering from Ewha Womans University in 2002 and 2006 respectively. And she received the Ph.D. degree in Computer Science from Seoul National University in 2013. From 2002 to 2004, she joined Shinhan Financial Group as a developer in applied research. In 2010, she was with U.C. Berkeley as a researcher. Since 2013, she has been with Soongsil University as a professor in the School of Electronic Engineering. Her research interests include social and computer networks, and security.