MSMA/CA: Multiple Access Control Protocol for Cognitive Radio-Based IoT Networks

Muhammad Shafiq, Jin-Ghoo Choi
Department of Information and Communication Engineering, Yeungnam University, Korea
shafiq.pu@gmail.com, jchoi@yu.ac.kr

Abstract

In this paper, we propose a new MAC protocol for Cognitive Radio (CR)-based IoT networks, called MSMA/CA. We extend the standard CSMA/CA, adopted in IEEE 802.11 WLANs, to the CR networks with the minimal modification since it works well in the real world. We resolve the classical hidden/exposed terminal problems by a variant of RTS/CTS mechanism and further, the hidden primary terminal problem by the mutual spectrum sensing at the transmitter and the receiver. We also modify the backoff process of CSMA/CA to incorporate the blocking of secondary transmitters, with the aim of protecting ongoing primary transmissions from aggressive secondary users. We analyze the throughput and delay of our proposed scheme using the Markov chain model on the backoff procedure, and verify its accuracy by simulations. Simulation results demonstrate that our protocol is suitable for IoT networks since the performance is insensitive to the number of users or devices.

Keywords: Internet-of-Things, Cognitive radio network, Multiple access control, CSMA/CA

1 Introduction

Recently we have observed the emergence of Internet-of-Things (IoT) networks, where tremendous number of various devices like sensors, actuators and RFID tags can create, store and communicate information each other automatically [1]. Just two decades ago, we could not imagine millions of smart objects are connected to provide useful services for human. Currently, such scenarios are becoming real due to the development of tiny low-power and low-cost devices. With these smart devices, the IoT technology will bring a wide range of new applications such as smart homes, smart cities, smart grids, eHealth, etc [2-5].

The number of devices connected to the Internet is increasing everyday. A report by Cisco anticipates that in 2020, there will be more than 50 billion devices in the Internet [6]. Rapid growth of the IoT drives the development of new communication protocols, technologies and more spectrum. The main issues on protocols and technologies span the high throughput, low delay, low power consumption, high mobility and scalability [7]. Unfortunately, most of the promising communication technologies such as ZigBee, WiFi, 6LoWPAN, Bluetooth LE, etc. rely on the license-free Industrial, Scientific and Medical (ISM) bands for the spectrum [8-9]. Proliferation of smart devices will make the ISM bands congested and hence, we need more spectrum resources available.

In [10-12], the authors propose to use the Cognitive Radio (CR) technology for the IoT networks in order to resolve the spectrum issue. The CR technology enables the efficient utilization of radio spectrum through the dynamic spectrum access. The unlicensed Secondary Users (SUs) therein can exploit the wireless channel whenever it is not used by the licensed Primary Users (PUs). If the SUs can leave the channel at the instant of PUs' comeback, all the users can share the channel while ensuring the priority of PUs. The IoT devices can act as the SUs in the CR-based networks.

There are two kinds of architectures in CR networks, i.e. centralized networks and distributed networks. Centralized networks require the infrastructure like a basestation such that the central node manages the spectrum usage of SUs. On the other hand, in distributed networks, each SU coordinates the spectrum access in the ad-hoc fashion without any help of infrastructure. The distributed CR network is more suitable for the IoT environment due to the low cost, less complexity and facility of deployment. However, the lack of central nodes causes many challenges in distributed CR networks like the hidden primary terminal problem [13-15].

In previous literature, several MAC protocols have been proposed to resolve the hidden primary terminal problem in distributed CR networks [16-18]. Especially, in [18], the transmitter sends a control packet, called Prepare-To-Sense (PTS), to initiate the mutual spectrum sensing with the receiver. When both the transmitter and the receiver confirm all the neighboring PUs to be silent, they can transmit a data packet following the Ready-To-Send (RTS)/Clear-To-Send (CTS) procedure as in the standard CSMA/CA

*Corresponding Author: Jin-Ghoo Choi; E-mail: jchoi@yu.ac.kr
DOI: 10.3966/160792642019012001028

[19]. However, this scheme has a downside of the large overhead due to the PTS control packet.

In this paper, we design a new MAC protocol for CR- based IoT networks by modifying the existing CSMA/CA protocol in the minimal. Our protocol is called the Mutual Sense Multiple Access with Collision Avoidance (MSMA/CA) since it features the mutual spectrum sensing in the transmitter and the receiver as in [18]. Moreover, we incorporate the blocking mechanism of secondary transmitters in the backoff procedure of CSMA/CA. The blocking of SUs is common in CR networks since it reduces the interference to active PUs [20].

The contribution of this paper can be summarized as follows.

- We propose a simple MAC protocol for CR-based IoT networks by extending the standard CSMA/CA, which resolves the hidden primary terminal problem as well as the classical hidden/exposed terminal problems.
- We suggest the blocking mechanism of secondary transmitters, embedded in the backoff procedure, to protect the PUs from the control packets of the secondary network.
- We analyze the throughput of our proposed MAC protocol mathematically by modifying the famous Markov chain on the backoff procedure in [21]. We also provide the analytic expression for the average packet transmission delay.

The rest of this paper is organized as follows. In Section 2, we describe the system model. We design the proposed MAC protocol in Section 3 and analyze it in Section 4. In Section 5, we verify the accuracy of our mathematical model through simulations, and investigate the performance of our MAC protocol for various parameters. The conclusions follow in Section 6.

2 System Model

We consider a secondary network with N SUs, which is located in the coverage of the primary network. Each SU indicates an IoT device, and the primary network is an arbitrary licensed wireless system around the devices. The primary and the secondary networks operate in the non-cooperative manner such that there are no communications between PUs and SUs. SUs can occupy the white spaces in time and frequency opportunistically whenever all the adjacent PUs are inactive. However, the SUs should vacate the channel immediately if the activity of PUs is detected. Each SU has its own set of PUs in the neighbor. When SU i conducts the spectrum sensing, the adjacent PUs can be inactive with probability $H_{0,i}$ and active with probability $H_{1,i}$, respectively.

The performance of CR networks significantly depends on the activity detection capability of SUs. Unfortunately, the spectrum sensing technology is not perfect and the users suffer from false alarm and misdetection. When a spectrum sensor misdetects the activity of PUs, it announces the channel to be idle while there exist ongoing communications in the primary network. Then the SU can transmit packets and incur interference to the active PUs. On the other hand, if the spectrum sensor sets off the false alarm, SUs lose the transmission opportunity and waste time. Especially, the carrier sense method of ordinary WLANs is vulnerable in the low SINR regime [22]. We hence need the advanced spectrum sensing techniques to protect the faraway PUs, such as energy detection, matched filter detection or cyclostationary features detection [23-24]. Such technologies improve the detection accuracy at the cost of increased sensing time. We denote the misdetection probability of SU i as α_i and its false alarm probability as β_i.

We assume the single channel system for simplicity. However, our proposed MAC protocol can be extended to multichannel systems with minor changes.

3 Proposed MAC Protocol

Our proposed MAC protocol, MSMA/CA, is a modification of the standard CSMA/CA for CR-based IoT networks. Hence, the two protocols share the feature of collision avoidance by the random backoff mechanism. Our scheme, however, adopts the spectrum sensing as well as the carrier sensing to protect the active PUs. Specifically, we use the Notify-To-Sense (NTS) / Acknowledge-To-Sense (ATS) procedure for the data transmission, in place of the RTS/CTS. The NTS/ATS mechanism enables the mutual spectrum sensing in the transmitter and the receiver in the midst of the two control packets.

We illustrate the typical packet transmission procedure under MSMA/CA in Figure 1, which can be described as follows. First, each SU with packets to send selects a backoff counter at random. Then the SU performs the carrier sensing to see whether the channel is idle or not. If the channel is clear for the DIFS interval[1], the SU decreases its backoff counter one by one for every idle backoff slot time (whose length is denoted as σ). In Figure 1, the backoff counter has been chosen as 4. If there is no transmission by other SUs until the backoff counter reaches 0, the transmitter broadcasts an NTS packet to initiate the spectrum sensing at the receiver. The NTS packet makes the neighboring SUs silent during the spectrum sensing by updating their Network Allocation Vectors (NAVs) [19]. Right after sending the NTS, the transmitter also conducts the spectrum sensing in conjunction with the

[1] This time interval is defined in the IEEE 802.11 standard for Wireless LANs [19].

receiver. If there are no active PUs around, it waits for the ATS packet from the receiver. When the receiver decodes the NTS packet successfully, assuming it is not blocked by the spectrum sensing, the receiver returns the ATS packet to the transmitter. The transmitter sends the DATA packet and the receiver replies by the ACK to confirm the successful packet delivery. We note that the transmitter does not always receive the ATS packet for the associated NTS. There can be multiple NTS packets transmitted at the same time. Then the collision occurs and the receiver cannot decode the NTS correctly. Even when the receiver decodes the NTS packet at success, it can be blocked by the spectrum sensing and does not return the ATS to the transmitter. The detailed operation is studied in the next section for the performance analysis.

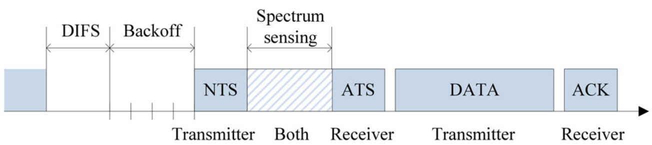


Figure 1. Typical packet transmission procedure

We now discuss the backoff procedure in detail. The primary objective of the backoff mechanism is to avoid the packet collisions in the secondary network. However, in the proposed scheme, the backoff process covers the blocking of the secondary transmitter as well when the transmitter detects the activity of adjacent PUs. For each data packet, the transmitter begins the backoff procedure from stage 0 with the contention window size W_0. Hence, the initial backoff counter is chosen among the integers from 0 to $W_0 - 1$. Once the counter decreases to 0 in the backoff stage $m(\geq 0)$, the SU attempts the packet transmission. If the transmission is successful, the backoff stage returns to 0 and the SU selects a new backoff counter in the interval $[0, W_0)$ at random. Otherwise, the backoff stage moves up to $m+1$, and the backoff counter should be chosen among $[0, W_{m+1})$. In our protocol, the selection probability of each backoff number depends on the cause of the transmission failure. When the transmission fails due to the transmitter's blocking, we choose the backoff counter among $[W_m, W_{m+1})$ rather than $[0, W_{m+1})$. This operation makes the SU be blocked for W_m slot times at least, and preserves the range of backoff counters. If not the case, we choose the backoff counter among $[0, W_{m+1})$ with the equal probability $\frac{1}{W_{m+1}}$.

There is a tradeoff between the channel utilization and the collision probability, according to the contention window size. Hence, as in the standard CSMA/CA, we adapt the contention window size depending on the number of competing SUs. Specifically, we double the window size whenever a packet transmission experiences consecutive failures and the backoff stage moves up. Thus the contention window size of stage m ($\geq$ 0) is given as where M denotes the maximum backoff stage. The transmitter can retransmit the failed packet for *retry limit* times at the maximum. The backoff process, however, remains in the stage M after the M-th trial on.

$$W_m = \min(2^m, 2^M) \cdot W_0. \quad \textbf{(1)}$$

The core of our protocol is the NTS/SS/ATS mechanism for the packet transmission, where SS denotes the spectrum sensing. It is designed to combine the benefits of the RTS/CTS mechanism and the spectrum sensing. In the standard CSMA/CA, the transmitter and receiver exchanges RTS/CTS packets to resolve the hidden/exposed terminal problems by updating the NAVs of neighboring users. Our NTS/ATS plays exactly the same function in the secondary network. Figure 2 illustrates the packet transmission from SU *i* to SU *j*, where the solid lines represent the Carrier Sense Range (CSR) and the dotted lines do the Spectrum Sense Range (SSR), respectively. The NTS packet from SU *i* keeps the exposed terminal *E* silent, in order for the transmitter to receive the ATS packet successfully. The ATS packet from SU *j* lets the hidden terminal *H* defer the transmission until the receiver sends the ACK back. Furthermore, the spectrum sensing in SU *j* detects the primary terminal PU, hidden from SU *i*, and stops the transmitter sending a data packet by holding the ATS packet. In this way, our protocol resolves the hidden primary terminal problem as well as the hidden/exposed terminal problems.

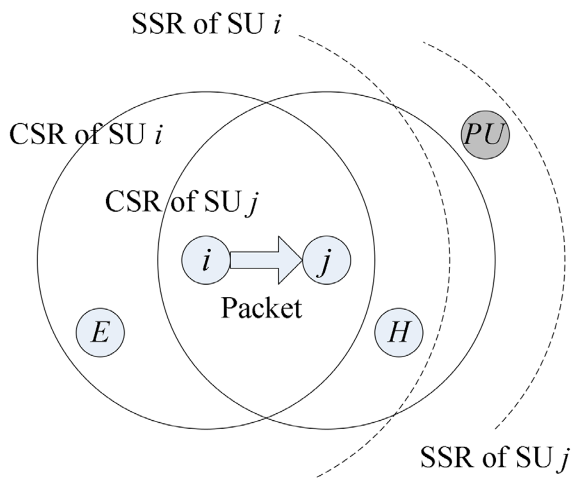


Figure 2. Packet transmission from SU *i* to SU *j*

4 Performance Analysis

In this section, we analyze the performance of our MAC protocol in terms of throughput and delay. We first summarize the notations used in the analysis in Table 1 for further references.

Table 1. Summary of notations

Notation	Description
$H_{1,i}(H_{0,i})$	Active (inactive) probability of the PUs around SU i
α_i	Misdetection probability of SU i
β_i	False alarm probability of SU i
C_i	Clear channel probability of SU i at arbitrary time
p_i	Probability of E_1 when SU i attempts the transmission
$q_{1,i}$	Probability of E_2 when SU i attempts the transmission
$q_{2,i}$	Probability of E_3 when SU i attempts the transmission
s_i	Probability of E_4 when SU i attempts the transmission
T_k	Length of an E_k slot (k = 1; 2; 3; 4)
Q_i	Probability of SU i's queue being backlogged
τ_k	Transmission attempt probability of SU i
n_{ij}	Packet transmission probability of SU i to SU j
W_m	Contention window size of the m-th backoff stage
M	Maximum backoff stage
q_i	Sum of probabilities $q_{1,i}$ and $q_{2,i}$
e_i	Failure probability of transmission attempts of SU i
$\pi_{m,b}$	Probability of the backoff counter being b in stage m
P_b	Occurrence probability of backoff slots
P_s	Occurrence probability of successful transmission slots
P_f	Occurrence probability of failed transmission slots
θ	Throughput of MSMA/CA
S	Length of an arbitrary slot
σ	Length of a backoff slot
T_s	Length of a successful transmission slot
T_f	Length of a failed transmission slot
HDR	Header size of a packet
L	Payload size of a packet
R	Channel transmission rate
D_i	Delay of a HOL packet of SU i
K_i	Number of slots a HOL packet of SU i sees
$d_{m,i}$	Average number of slots for SU i to stay at stage m

4.1 Packet Transmission Process

We observe the packet transmission process of our MSMA/CA. Each SU initiates the packet transmission when its backoff counter decreases to 0 and the packet queue is not empty. The transmission attempt encounters one of the four mutually exclusive events: (1) Blocking at the transmitter, (2) Collision of multiple NTS packets, (3) Blocking at the receiver and (4) Successful transmission. We denote each event as E_1, E_2, E_3 and E_4, respectively. We henceforth calculate the probability of the individual event and specify how much time the transmitter spends when each event happens. To this end, we define the *clear channel* probability C_i of an SU *i* as

$$C_i = \alpha_i H_{1,i} + (1-\beta_i) H_{0,i}.. \quad \textbf{(2)}$$

where we recall that α_i is the misdetection probability of SU *i*, β_i is its false alarm probability, $H_{0,i}$ is the inactive probability of the PUs around SU *i* and $H_{1,i}$ is the active probability of the PUs, respectively. Thus, when SU *i* performs the spectrum sensing, it decides the adjacent PUs to be inactive with probability C_i and active with probability $1-C_i$.

Suppose that SU *i* has the backoff counter 0 and broadcasts the NTS packet. The transmitter conducts the spectrum sensing right after changing the transceiver to the reception mode. Then SU *i* perceives the adjacent PUs active, which is event E_1, with probability

$$p_i = 1 - C_i. \quad \textbf{(3)}$$

When E_1 happens, the SU is blocked until the possible return time of the ATS packet (and blocked further by our proposed backoff process). As shown in Figure 3a, SU *i* spends T_1 time and resumes the packet transmission process then. We call the time interval from the broadcast of the NTS packet to the following DIFS time, as the E_1 slot. The length of E_1 slot is given as

$$T_1 = NTS + SS + SIFS + ATS + DIFS. \quad \textbf{(4)}$$

where *NTS*, *ATS* and *SS* represent the transmission time of an NTS, that of an ATS and the spectrum sensing time, respectively.

We consider an SU *i* that has transmitted the NTS packet and not perceived any PU activity during the spectrum sensing. The SU does not necessarily receive the ATS packet from the receiver since its NTS may have collided with other NTS packets, which is event E_2. Indeed the transmitter may not receive the ATS packet due to the blocking of the receiver, which is defined as a separated event, E_3 later. An SU *i*, which attempts the packet transmission, can experience the event E_2 with the probability

$$q_{1,i} = C_i \left(1 - \prod_{k \neq i} (1 - Q_k \tau_k)\right) \quad \textbf{(5)}$$

where Q_k is the probability of SU *k*'s queue being backlogged and τ_k denotes the transmission attempt probability of SU *k* (incurred by the zero backoff counter). It would be helpful to see that $1-Q_k\tau_k$ is the probability of SU *k* not trying the packet transmission. So, all the SUs other than SU *i* do not attempt the transmission with probability $\prod_{k \neq i}(1-Q_k\tau_k)$. We note that some of the transmitters can be blocked for ATS time due to the neighboring PUs' activity. We require the SU *i* to be synchronized with the other transmitters, which may be blocked. Hence, from Figure 3b, the length of E_2 slot is given as

$$T_2 = NTS + SS + SIFS + ATS + DIFS. \quad (6)$$

Suppose that SU i has sent the NTS packet and is waiting for the response from the corresponding SU j. Even though the SU j has received the NTS packet successfully, it cannot return the ATS if the SU is blocked by the spectrum sensing. That is the event E_3. When SU i sends the data packets to SU j with probability η_{ij}, we can calculate the probability of E_3 as

$$q_{2,i} = C_i \sum_{j \neq i} \eta_{ij} \left((1 - Q_j \tau_j)(1 - C_j) \prod_{k \neq i,j} (1 - Q_k \tau_k) \right). \quad (7)$$

We have let the non-blocked transmitter wait for ATS time in the event E_2. Similarly, when the receiver is blocked, it also waits for ATS time to keep pace with the transmitter. As shown in Figure 3c, the length of E_3 slot is given as

$$T_3 = NTS + SS + SIFS + ATS + DIFS. \quad (8)$$

Finally, we consider the successful packet transmission case. SU i sends the NTS packet and after finding the PUs inactive, it waits for the ATS from the receiver SU j. If the SU j is not blocked, it returns the ATS packet. Then SU i transmits a DATA packet and SU j responds by the ACK always since we assume the error-free channel. For SU i, the occurrence probability of E_4 is given as

$$s_i = C_i \sum_{j \neq i} \eta_{ij} \left((1 - Q_j \tau_j) C_j \prod_{k \neq i,j} (1 - Q_k \tau_k) \right). \quad (9)$$

Figure 3d shows that the length of E_4 slot, i.e. the time interval for the successful packet transmission, can be written as

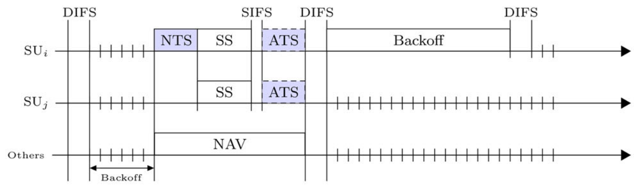
$$\begin{aligned} T_4 = {} & NTS + SS + SIFS + ANTS + SIFS + DATA \\ & + SIFS + ACK + DIFS \end{aligned} \quad (10)$$

where DATA and ACK denote the transmission time of a DATA and that of an ACK, respectively.

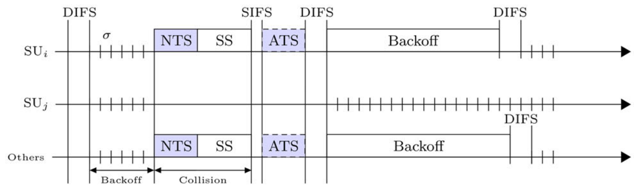
4.2 Throughput

We analyze the performance of our proposed MAC protocol under the following assumptions.

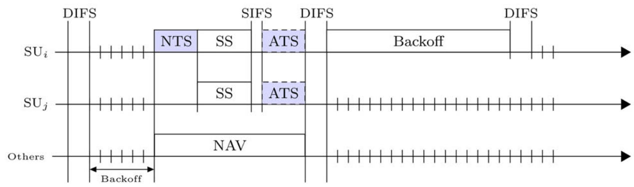
- The secondary network is a single hop such that an SU can transmit packets to other SUs directly.
- Each SU has the packets to send at all times, i.e. $Q_i = 1$ for every SU i.
- An SU retransmits a failed data packet until it is delivered successfully. In other words, we set the retry limit to infinite.
- When an SU attempts a transmission, it succeeds with a constant probability regardless of the current backoff stage or the number of SUs.



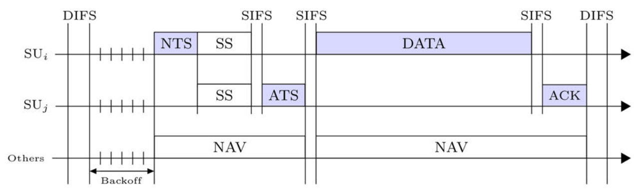
(a) Blocking at the transmitter



(b) Collision of multiple NTS packets



(c) Blocking at the receiver



(d) Successfull transmission.

(d) Successful transmission

Figure 3. Events following the packet transmission attempt

We observe the backoff process of an arbitrary SU i at the beginning of each slot. Every slot, the backoff counter decreases one by one and when it reaches 0, SU i starts the packet transmission process. If the transmission is successful, SU i returns to the initial backoff stage and otherwise, its backoff stage moves up one step. If the SU i has encountered E_1 at stage $m \in [0, M)$, the new backoff counter is chosen among $[W_m, W_{m+1})$ with the equal probability. On the other hand, if the SU has suffered either E_2 or E_3, its backoff counter is chosen among $[0, W_{m+1})$.

We model the backoff process as a discrete time Markov chain with two dimensional states (m,b), where $m \in [0, M]$ is the backoff stage and $b \in [0, W_M)$ is the backoff counter in the stage m. We illustrate the Markov chain in Figure 4, where $q_i = q_{i,1} + q_{2,i}$ and the subscript i can be dropped without confusion. We also use the notation $e_i = p_i + q_i$ henceforth to denote the transmission error (or failure) probability of SU *i*. The state transition probabilities of our Markov chain can be written as eq. (11). As the Markov chain is constructed, we can set up the detailed balance equation for each state (m,b), and calculate the steady state probability $\pi_{m,b}$ by solving the system of equations. Then, the transmission attempt probability of an SU *i* can be written as

$$\tau_i = \left(\frac{2(1+p_i)(1-e_i)-(2e_i+p_i)(2e_i)^M}{4(1-2_{e_i})} W_0 + \frac{1}{2} \right)^{-1} \quad \textbf{(12)}$$

See the Appendix for the derivation of τ_i.

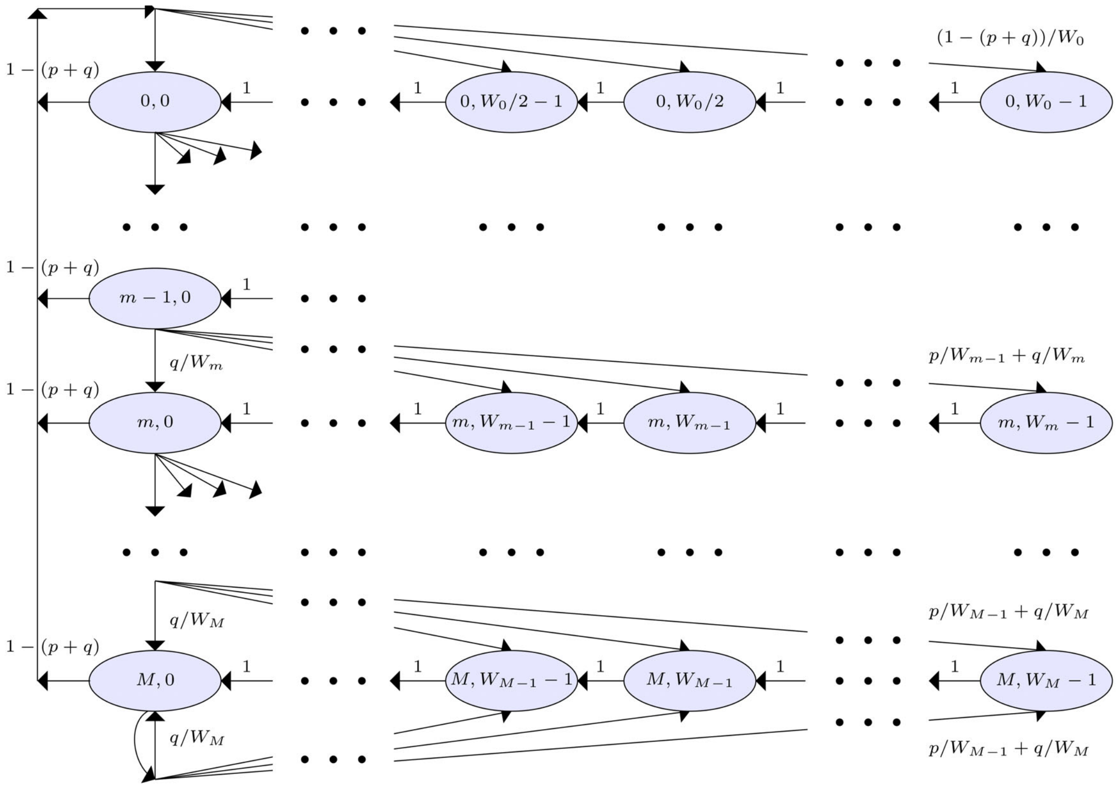


Figure 4. Markov chain model of the backoff process

$$\begin{cases} \Pr\{(m,b) \to (m,b-1)\} = 1 & \text{for } m \in [0,M], b \in [1, W_m) \\ \Pr\{(m,0) \to (0,b)\} = \dfrac{1-(p+q)}{W_0} & \text{for } m \in [0,M], b \in [0, W_0) \\ \Pr\{(m,0) \to (m+1,b)\} = \dfrac{q}{W_m+1} & \text{for } m \in [0,M], b \in [0, W_m) \\ \Pr\{(m,0) \to (m+1,b)\} = \dfrac{p}{W_m} + \dfrac{q}{W_{m+1}} & \text{for } m \in [0,M], b \in [W_m, W_{m+1}) \\ \Pr\{(M,0) \to (M,b)\} = \dfrac{q}{W_M} & \text{for } b \in [0, W_{M-1}) \\ \Pr\{(M,0) \to (M,b)\} = \dfrac{p}{W_{M-1}} + \dfrac{q}{W_M} & \text{for } b \in [W_{M-1}, W_m). \end{cases} \quad \textbf{(11)}$$

We can classify the slots into the backoff slot and the event slots (i.e., E_1, E_2, E_3 or E_4). The backoff slots occurs when no SU attempts to transmit packets, whose occurrence probability is written as

$$P_b = \prod_i (1-\tau_i). \quad \textbf{(13)}$$

We know that SU *i* transmits a data packet successfully when its backoff counter becomes 0 and the event E_4 happens, whose probability is given as $\tau_i s_i$. So, we have the slots with the successful data transmission with probability

$$P_s = \sum_i \tau_i s_i. \quad \textbf{(14)}$$

The other slots correspond to the failed packet transmissions. Such slots are found with probability

$P_f = 1-(P_b + P_s)$.

The throughput of our proposed scheme can be written as

$$\theta = \frac{\text{Average payload bits transmitted in a slot}}{\text{Average slot length}}. \quad \textbf{(15)}$$

We obtain the average of the slot length S as

$$E[S] = P_b\sigma + P_s E[T_s] + P_f E[T_f]. \quad \textbf{(16)}$$

where $T_s = T_4$ and $T_f = T_1 (= T_2 = T_3)$. From eq. (10), we can see that

$$\begin{aligned} E[T_s] &= NTS + SS + ATS + E[DATA] \\ &\quad + ACK + 3SIFS + DIFS \\ &= NTS + SS + ATS + (HDR + E[L])/R \\ &\quad + ACK + 3SIFS + DIFS \end{aligned} \quad \textbf{(17)}$$

where HDR, L and R denote the header size, payload size and transmission rate, respectively. From eq. (4), we know that T_1 is fixed and thus $E[T_f] = T_1$. Conclusively, the throughput of our protocol is given as

$$\theta = \frac{P_s E[L]}{P_b\sigma + P_s E[T_s] + P_f T_1}. \quad \textbf{(18)}$$

4.3 Delay

We now analyze the average delay of a data packet on the condition that the packet transmission is successful. In SU i, each Head-Of-Line (HOL) packet at the queue sees K_i slots until the successful transmission, through the backoff process. Then the packet delay D_i can be written as

$$D_i = \sum_{k=1}^{K_i} S_k + T_s. \quad \textbf{(19)}$$

where S_k denotes the length of the k-th slot the packet observes. So, the average packet delay is given as

$$\begin{aligned} E[D_i] &= E\left[\sum_{k=1}^{K_i} S_k\right] + E[T_s] \\ &= E[K_i]\cdot E[S] + E[T_s] \end{aligned} \quad \textbf{(20)}$$

where we remove the subscript k of S_k without loss of generality. We have obtained the average slot length $E[S]$ already in eq. (16) and the average successful transmission time $E[T_s]$ in eq. (17). We hence need to calculate the average number of backoff slots, $E[K_i]$, to the successful transmission.

Each packet transmission of SU i begins from the backoff stage 0. It stays in stage 0 for $d_{0,i} = \sum_{b=0}^{W_0-1} b\frac{1}{W_0} = \frac{W_0-1}{2}$ slot times in average until the transmission attempt. If the SU i transmits a packet at stage $m-1 \in (0, M]$ and does not succeed, the backoff stage moves up to m and the backoff counter is chosen among 0 to $W_m - 1$. We recall that each number can be chosen as the backoff counter with different probabilities, depending on the cause of the transmission failure. Once SU i enters the backoff stage $m \in (0, M]$, it stays there in average for

$$\begin{aligned} d_{m,i} &= \sum_{b=0}^{W_{m-1}-1} b\frac{q_i}{W_m} + \sum_{b=W_{m-1}}^{W_m-1} b\left(\frac{p_i}{W_{m-1}} + \frac{q_i}{W_m}\right) \\ &= \frac{3W_{m-1}-2}{4} + \frac{W_{m-1}-1}{2}q_i \\ &= \frac{4e_i - p_i}{8}W_m - \frac{e_i}{2}. \end{aligned} \quad \textbf{(21)}$$

For the third equality, we recall $W_m = 2W_{m-1}$ and $e_i = p_i + q_i$. Furthermore, we let $d_{m,i} = d_{M,i}$ for $m \geq M+1$ since the backoff process is in stage M during the M-th or more retrials.

If a data packet is transmitted successfully at backoff stage m, whose probability is $e_i^m(1-e_i)$, it has passed through $\sum_{n=0}^{m} d_{n,i}$ slots in average. So, $E[K_i]$ can be written as

$$\sum_{m=0}^{\infty} e_i^m (1-e_i)\left(\sum_{n=0}^{\infty} d_{n,i}\right) = (1-e_i)\sum_{n=0}^{\infty} d_{n,i} \sum_{m=n}^{\infty} e_i^m = \sum_{n=0}^{\infty} e_i^n d_{n,i}. \quad \textbf{(22)}$$

Hence, the number of slots to the successful transmission is given as

$$\begin{aligned} E[K_i] &= \sum_{n=0}^{M} e_i^n d_{n,i} + \sum_{n=M+1}^{\infty} e_i^n d_{n,i} \\ &= \sum_{n=0}^{M} e_i^n \left(\frac{4e_i - p_i}{8} 2W_0 - \frac{e_i}{2}\right) + \sum_{n=M+1}^{\infty} e_i^n d_{M,i} \\ &= \frac{4e_i - p_i}{8}\left(\frac{1-(2e_i)^{M+1}}{1-2e_i} + \frac{e_i(2e_i)^M}{1-e_i}\right)W_0 - \frac{e_i}{2(1-e_i)}. \end{aligned} \quad \textbf{(23)}$$

For the second equality, we replaced $d_{n,i}$ with $\frac{4e_i - p_i}{8}W_n - \frac{e_i}{2}$ for $n \in [0, M]$ from eq. (21) and for $n \geq M+1$, $d_{n,i}$ is fixed as $d_{M,i}$. The third equality follows after some algebra and rearrangement of terms.

5 Simulation Results

We have developed the simulation code with the C++ language, considering all the details of our MSMA/CA protocol. We assume that each SU i has the ideal spectrum sensing capability, i.e. $\alpha_i = 0$ and $\beta_i = 0$. Further, the channel is assumed error-free such that the transmission error does not occur. We summarize the default parameter values used in the simulations in Table 2. Note that the activity rate of PUs, $H_{1,i}$ is set equal for every SU i such that each SU experiences the identical performance in the proposed scheme. So, in this section, we drop the subscript i denoting SU i.

Table 2. Default parameters in simulations

Parameter name	Value
PHY header	120 bits
MAC header	272 bits
Payload	8184 bits
NTS	160 bits + PHY header
ATS (and ACK)	112 bits + PHY header
SIFS time	10 μs
DIFS time	50 μs
Backoff slot time (σ)	20 μs
PU activity rate ($H_{1,i}$)	0.01
Spectrum sensing time	0.5 ms
Channel transmission rate (R)	1 Mbps
Initial CW size (W_0)	32
Maximum CW size (W_M)	1024
Maximum backoff stage (M)	5

We have run each simulation 1000 times and averaged them to obtain one simulation result. We compare the

analysis results (shown in lines) with the simulation results (shown in markers) in Figure 5 and Figure 6. We can observe that our analysis is very accurate since the two results match each other closely.

Figure 5(a) shows the throughput, normalized by the channel transmission rate R, of our proposed scheme while varying the number of SUs, N. We consider the minimum contention window size of 32, 64 and 128, respectively. As the number of SUs increases, the normalized throughput increases first and then decreases monotonically. When the number of SUs is small, the channel time is wasted by backoff slots since the smallest backoff counter chosen by SUs becomes large. Then the system has low throughput. On the other hand, if the number of SUs is large, there happen frequent collisions and the throughput decreases. Further, we observe that the large initial contention window size is better for many SUs since it reduces the collision rate significantly.

In Figure 5(b), we demonstrate the effect of PU activity rate $H_{1,i}$ on the normalized throughput. We consider the number of SUs of 10, 30 and 50, respectively. Regardless of the number of SUs, the throughput decreases monotonically as $H_{1,i}$ increases since the available channel time decreases. When the PU activity rate is low, the throughput is slightly better for small N since the collisions do not happen often. Conversely, when the PU activity rate is high, the throughput is larger with large N. It can be explained as follows. If the PUs are active with high probability, many SUs are liable to be blocked after the spectrum sensing and cannot participate in the packet transmission. Hence the overall throughput decreases.

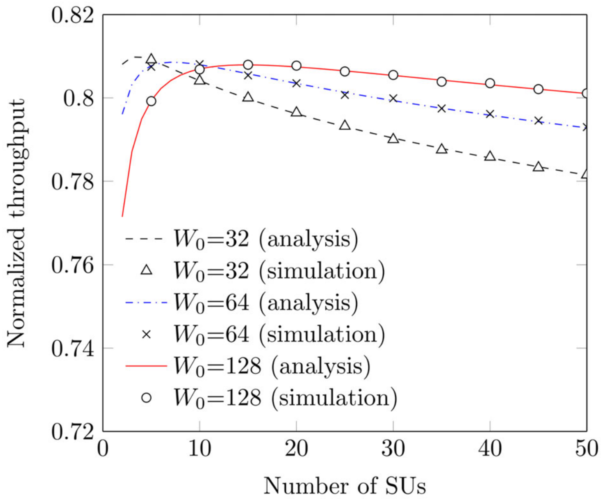
We provide the normalized throughput for various numbers of SUs in Figure 5(c), where we consider the maximum backoff stage M of 3, 5 and 7, respectively. The throughput is not sensitive to the M when the number of SUs is small since the collisions do not happen often. However, when the number of SUs is large, the throughput decreases fast with the small M since the range of backoff counters does not increase enough while collisions occur frequently.

In Figure 6, we show the transmission attempt probability τ while increasing the number of SUs. We can see that τ decreases with the number of SUs since the frequent collisions make the backoff counters increase. We also observe that when the initial contention window size is large, the transmission attempt probability decreases even though the collisions occur infrequently since the initial backoff counters are large.

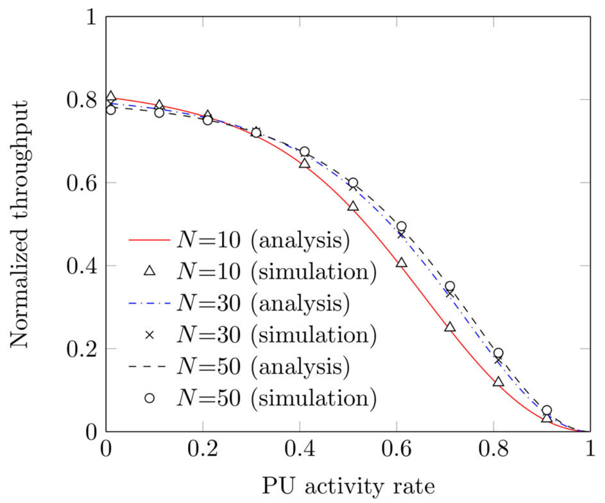
We investigate the average delay D of the successfully delivered packets in Figure 7. The delay increases as the number of SUs increases for given initial contention windows size W_0 (in Figure 7(a)) and maximum backoff stage M (in Figure 7(b)). When there are many SUs, each SU experiences frequent collisions and its backoff counter becomes large due to the consecutive transmission failures, which leads to the large transmission delay. We observe that when W_0 (or M) is small, the average delay decreases even with more collisions since the backoff counters chosen are small.

6 Conclusion

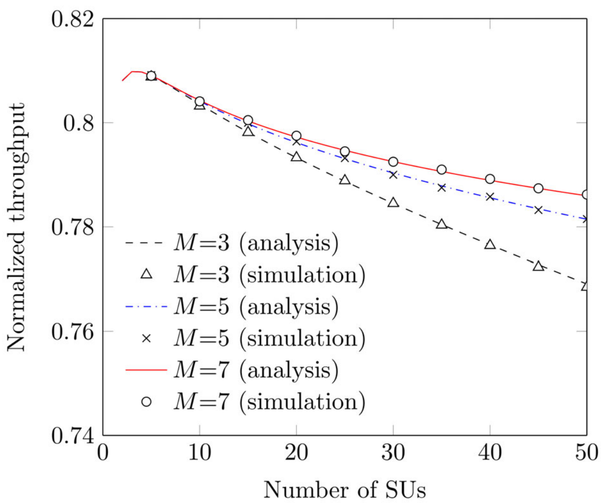
We have proposed a new MAC protocol, called MSMA/CA, for CR-based IoT networks. Our scheme combines the RTS/CTS mechanism of the standard CSMA/CA with the mutual spectrum sensing technique of CR networks. As a result, our MAC protocol resolves the classical hidden/exposed terminal problems as well as the hidden primary terminal problem at the same time. We also modify the backoff process of CSMA/CA in order to incorporate the blocking mechanism of SUs. The SUs hold the packet transmission when they detect the active PUs during the spectrum sensing, by choosing a large backoff counter. Hence the PUs are protected from the persistent transmission attempts of SUs. We have analyzed the proposed protocol and obtained the throughput and delay performances. Simulation results show that our analysis is accurate for various scenarios. Our scheme is a good candidate for the MAC protocol of CR-based IoT networks since the performance is insensitive to the number of users (or IoT devices) while providing sufficient priority to the incumbent PUs.



(a) Normalized throughput vs. number of SUs (for various W_0 s)



(b) Normalized throughput vs. PU activity rate



(c) Normalized throughput vs. number of SUs (for various Ms)

Figure 5. Normalized throughput

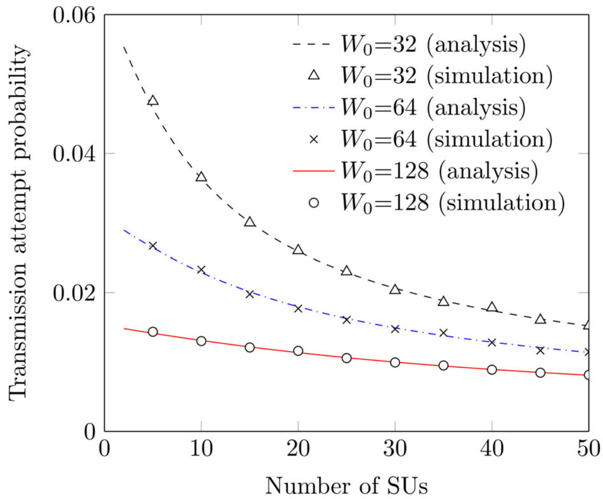
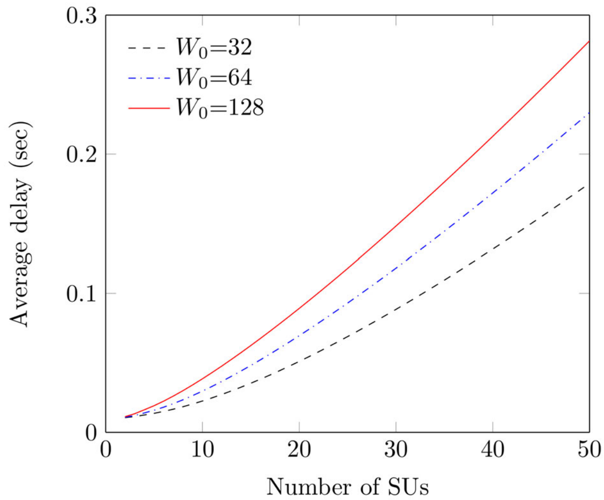
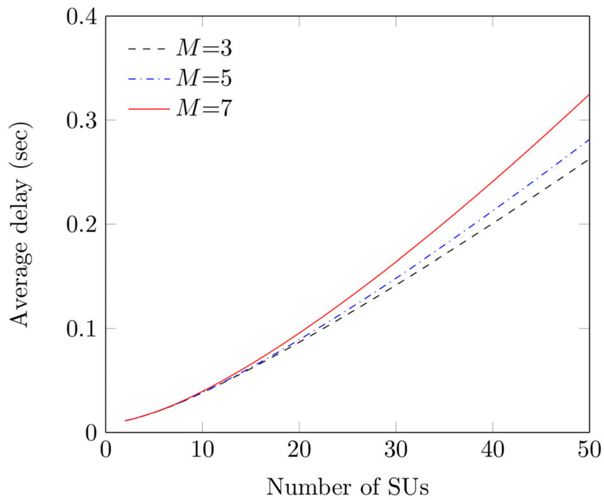


Figure 6. Transmission attempt probability vs. number of SUs



(a) Effect of the initial contention window size W_0



(b) Effect of the maximum backoff stage M

Figure 7. Average delay vs. number of SUs

Acknowledgments

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2017-2016-0-00313) supervised by the IITP (Institute for Information & communications Technology Promotion). J.-G. Choi is the corresponding author.

References

[1] B. Rajkumar, A.V. Dastjerdi, *Internet of Things: Principles and Paradigms*, Elsevier, 2016.

[2] M. R. Palattella, M. Dohler, A. Grieco, G. Rizzo, J. Torsner, T. Engel, L. Ladid, Internet of Things in the 5G Era: Enablers, Architecture and Business Models, *IEEE Journal on Selected Areas in Communications*, Vol. 34, No. 3, pp. 510-527, 2016.

[3] A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi, Internet of Things for Smart Cities, *IEEE Internet of Things Journal*, Vol. 1, No. 1, pp. 22-32, 2014.

[4] C. Perera, C. H. Liu, S. Jayawardena, The Emerging Internet of Things Marketplace from an Industrial Perspective: A Survey, *IEEE Transactions on Emerging Topics in Computing*, Vol. 3, No. 4, pp. 585-598, 2015.

[5] D. V. Dimitrov, Medical Internet of Things and Big Data in Healthcare, *Healthcare Informatics Research*, Vol. 22, No. 3, pp. 156-163, 2016.

[6] D. Evans, *The Internet of Things: How the Next Evolution of the Internet is Changing Everything*, Cisco Internet Business Solutions Group, April, 2011.

[7] S. Li, L.D. Xu, S. Zhao, The Internet of Things: A Survey, *Information Systems Frontiers*, Vol. 17, No. 2, pp. 243-259, 2015.

[8] A. Fuqaha, A. Guizani, M. Mohammadi, M. Aledhari, M. Ayyash, Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications, *Communications Surveys and Tutorials*, Vol. 17, No. 4, pp. 2347-2376, 2015.

[9] M. Elkhodr, S. Shahrestani, H. Cheung, Emerging Wireless Technologies in the Internet of Things: a Comparative Study, *International Journal of Wireless and Mobile Networks* (IJWMN), Vol. 8, No. 5, pp. 67-82, 2016.

[10] R. Priyanka, K. D. Singh, J. M. Bonnin, Cognitive Radio for M2M and Internet of Things: A Survey, *Computer Communications*, Vol. 94, pp. 1-29, 2016.

[11] K. A. Ali, M. H. Rehmani, A. Rachedi, When Cognitive Radio meets the Internet of Things, *Proceedings of EEE Wireless Communications and Mobile Computing Conference (IWCMC)*, Cyprus, 2016, pp. 469-474.

[12] S. M. Ali, S. Zhang, C. Maple, Cognitive Radio Networks for Internet of Things: Applications, Challenges and Future, *Proceedings of the International Conference on Automation and Computing*, London, UK, 2013, pp.13-14.

[13] G. Ganesan, Y. Li, Cooperative Spectrum Sensing in Cognitive Radio Networks, *Proceedings of IEEE Int. Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Baltimore, MD, 2005, pp. 137-143.

[14] D. Cabric, A. Tkachenko, R. Brodersen, Spectrum Sensing Measurements of Pilot, Energy, and Collaborative Detection, *Proceeding of IEEE Military Communications Conference*, Washington, DC, 2006, pp. 1-7.

[15] G. Ganesan, Y. Li, Agility Improvement through Cooperative Diversity in Cognitive Radio, *Proceedings of IEEE Global Communications Conference*, Vol. 5, St. Louis, MO, 2005, pp. 2505-2509.

[16] Q. Zhao, L. Tong, A. Swami, Y. Chen, Decentralized Cognitive MAC for Opportunistic Spectrum Access in Ad Hoc Networks: A POMDP Framework, *IEEE Journal on Selected Areas in Communications*, Vol. 25, No. 3, pp. 589-600, 2007.

[17] S. C. Jha, U. Phuyal, M. M. Rashid, V. K. Bhargava, Design of OMCMAC: An Opportunistic Multi-channel MAC with QoS Provisioning for Distributed Cognitive Radio Networks, *IEEE Transactions on Wireless Communications*, Vol. 10, No. 10, pp. 3414-3425, 2011.

[18] Q. Chen, W. Wong, M. Motani, Y. Liang, MAC Protocol Design and Performance Analysis for Random Access Cognitive Radio Networks, *IEEE Journal on Selected Areas in Communications*, Vol. 31, No. 11, pp. 2289-2298, 2013.

[19] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std 802.11-2007 Part 11, 2007.

[20] A. M. Wyglinski, M. Nekovee, T. Hou, *Cognitive Radio Communications and Networks: Principles and Practice*, Academic Press, 2009.

[21] G. Bianchi, Performance Analysis of the IEEE 802.11 Distributed Coordination Function, *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 3, pp. 535-547, 2000.

[22] L. Wei, O. Tirkkonen, Spectrum Sensing in the Presence of Multiple Primary Users, *IEEE Transactions on Communications*, Vol. 60, No. 5, pp. 1268-1277, 2012.

[23] D. T. C. Wong, S. Zheng, Y. Liang, Cognitive Multi-Channel MAC Protocols with Perfect and Imperfect Sensing, *Proceeding of IEEE International Conference on Communications*, Kyoto, Japan, 2011.

[24] Y. Tevfik, H. Arslan, A Survey of Spectrum Sensing Algorithms for Cognitive Radio Applications, *Communications Surveys and Tutorials*, Vol. 11, No.1, pp. 116-130, 2009.

Biographies

Muhammad Shafiq is a Ph.D. candidate in Department of Information and Communication Engineering, Yeungnam University, Korea. He received a M.S. degree in Computer Science from UIIT., PMAS., Arid Agriculture University, Rawalpindi, Pakistan. He also received a master's degree in Information Technology (IT) from University of the Punjab, Gujranwala, Pakistan. He has been working with the Faculty of Computing and IT University of Gujrat, Gujrat, Pakistan as a faculty member since 2010. He was with the Department of Computer Science and IT, Federal Urdu University, Islamabad, Pakistan in 2009. His research interests include the design of spectrum management, routing, and medium access control protocols for Mobile Ad Hoc Networks, Internet of Things, and Cognitive Radio Networks.

Jin-Ghoo Choi received his Ph.D. degree from the School of Electrical Engineering and Computer Science, Seoul National University in 2005. From 2006 to 2007, he worked for Samsung Electronics as a senior engineer. In 2009, he was with the Department of Electrical and Computer Engineering in The Ohio State University as a visiting scholar. He joined the Department of Information and Communication Engineering in Yeungnam University as a faculty member in 2010. His research interests include performance analysis of communication networks, resource management in wireless networks, and wireless sensor network.

Appendix

Derivation of τ_i : We derive the transmission attempt probability τ_i of SU i by analyzing the Markov chain of Figure 4. For simplicity, we here drop the subscript i that denotes SU i.

We first list up the detailed balance equation at each state (m,b) as follows. Defining $\tau = \sum_{m=0}^{M} \pi_{m,0}$, we can write the balance equations of the backoff stage 0 as

$$\begin{aligned} &\pi_{0,b} = \pi_{0,b+1} + \frac{1-(p+1)}{W_0}\tau \text{ for } b \in [0, W_0 - 1), \\ &\pi_0, W_0 - 1 = \frac{1-(p+q)}{W_0}\tau. \end{aligned} \tag{24}$$

The balance equations of stage $m \in [1, M-1)$ are given as

$$\begin{aligned} &\pi_{m,b} = \pi_{m,b+1} + \pi_{m-1,0}\frac{q}{W_m} \text{ for } b \in [0, W_{m-1}), \\ &\pi_{m,b} = \pi_{m,b+1} + \pi_{m-1,0}\frac{2p+q}{W_m} \text{ for } b \in [W_{m-1}, W_{m-1}), \\ &\pi_m W_m - 1 = \pi_{m-1,0}\frac{2p+q}{W_m}. \end{aligned} \tag{25}$$

Lastly, the balance equations for stage M are as follows.

$$\begin{aligned} &\pi_{M,b} = \pi_{M,b+1} + (\pi_{M-1,0} + \pi_{M,0})\frac{q}{W_M} \text{ for } b \in [0, W_{M-1}), \\ &\pi_{M,b} = \pi_{M,b+1} + (\pi_{M-1,0} + \pi_{M,0})\frac{2p+q}{W_M} \text{ for } b \in [W_{M-1}, W_{M-1}), \\ &\pi_M W_M - 1 = (\pi_{M-1,0} + \pi_{M,0})\frac{2p+q}{W_M}. \end{aligned} \tag{26}$$

We sum up the equations in (24) to obtain

$$\sum_{b=0}^{W_0-1} \pi_{0,b} = \sum_{b=1}^{W_0-1} \pi_{0,b} + (1-(p+q))\tau. \tag{27}$$

which leads to

$$\pi_{0,0} + (1-(p+q))\tau. \tag{28}$$

In the same way, by summing up the eqs. in (25), we have

$$\pi_{m,0} = (p+q)\pi_{m-1,0} = (p+q)^m \pi_{0,0}. \tag{29}$$

for $m = 1, \cdots, M-1$ and, from the eqs. in (26), we obtain $\pi_{M,0} = (p+q)(\pi_{M-1,0} + \pi_{M,0})$, i.e.

$$\pi_{M,0} = \frac{p+q}{1-(p+q)}\pi_{M-1,0} = \frac{(p+q)^M}{1-(p+q)}\pi_{0,0}. \tag{30}$$

We next observe the eqs. of (24) and by the back substitution, obtain the steady state probabilities for the stage 0 as

$$\begin{aligned} \pi_{0,b} &= \frac{1-(p+q)}{W_0}\tau(W_0 - b) \\ &= \pi_{0,0}\frac{W_0 - b}{W_0} \quad \text{for } b \in [0, W_0). \end{aligned} \tag{31}$$

For the stage $m \in [1, M)$, we have the steady state probabilities from eq. (25) as

$$\pi_{m,b} \begin{cases} \pi_{m-1,0}\left(p + \dfrac{q}{W_m}(W_m - b)\right) \text{ for } b \in [0, W_m - 1) \\ \pi_{m-1,0}\dfrac{2p+q}{W_m}(W_m - b) \text{ for } b \in [W_m - 1, W_m). \end{cases} \tag{32}$$

Further, from eq. (26), we have $\pi_{M,b}$ as

$$\begin{cases} (\pi_{M-1,0} + \pi_{M,0})\left(p + \dfrac{q}{W_m}(W_m - b)\right) \text{ for } b \in [0, W_{m-1}) \\ (\pi_{M-1,0} + \pi_{M,0})\dfrac{2p+q}{W_m}(W_m - b) \text{ for } b \in [W_{M-1}, W_m). \end{cases} \tag{33}$$

We now calculate the probability of the backoff process being in each individual stage. From eq. (31), the probability of stage 0 is given as

$$\pi_0 = \sum_{b=0}^{W_0-1} \pi_{0,b} = \pi_{0,0}\frac{W_0+1}{2}. \tag{34}$$

For stage $m \in [1, M-1)$, from eq. (32), we have

$$\pi_m = \sum_{b=0}^{W_m-1} \pi_{m,b} = \frac{\pi_{m-1,0}}{2}\left(\frac{3p+2q}{2}W_m + (p+q)\right). \tag{35}$$

The result for the last stage M is obtained from eq. (33) as

$$\pi_M = \sum_{b=0}^{W_M-1} \pi_{M,b} = \frac{\pi_{M-1,0} + \pi_{M,0}}{2}\left(\frac{3p+2q}{2}W_M + (p+q)\right). \tag{36}$$

We note that $\sum_{m=0}^{W_M} \pi_M = 1$ since the total steady state probability should be 1. Using $e = p+q$, we have the following equation.

$$\begin{aligned} &\pi_0 + \sum_{m=1}^{M-1}\pi_m + \pi_M \\ &= \pi_0 + \sum_{m=1}^{M-1}\frac{\pi_{m-1,0}}{2}\left((e+\frac{p}{2})W_m + e\right) \\ &+ \frac{\pi_{M-1,0}}{2}\left((e+\frac{p}{2})W_m + e\right) + \frac{\pi_{M,0}}{2}\left((e+\frac{p}{2})W_M + e\right) \\ &= \pi_0 + \frac{\pi_{0,0}}{2}\sum_{m=1}^{M} e^{m-1}\left((e+\frac{p}{2})W_m + e\right) + \frac{\pi_{M,0}}{2}\left((e+\frac{p}{2})W_m + e\right) \\ &= \frac{\pi_{0,0}}{2}(W_0+1) + \frac{\pi_{0,0}}{2}\left((2e+p)\frac{1-(2e)^M}{1-2e}W_0 + \frac{e(1-e)^M}{1-e}\right) \\ &= 1. \end{aligned} \tag{37}$$

Recalling $\tau = \dfrac{\pi_{0,0}}{1-e}$ from eq. (28), we have the transmission attempt probability τ of eq. (12) after rearranging the terms in eq. (37).