Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 3096710, 15 pages
https://doi.org/10.1155/2018/3096710

Research Article

Acknowledgement Corruption: A New Aspect of Physical Layer Capture in IEEE 802.11 Networks

Jung-Han Han, Hyoung-Gyu Choi, and Seung-Jae Han

Department of Computer Science, Yonsei University, Seoul, Republic of Korea

Correspondence should be addressed to Seung-Jae Han; seungjaehan@yonsei.ac.kr

Received 28 June 2017; Revised 10 December 2017; Accepted 9 January 2018; Published 15 February 2018

Academic Editor: Gianluigi Ferrari

Even if a collision occurs in IEEE 802.11 network, a transmission may be successfully decoded at the receiver if the signal strength of one transmission is sufficiently stronger than the other transmission. This phenomenon is called "Physical Layer Capture" (PLC). While existing works have considered PLC between data frames, in this paper we investigate the case that an ACK frame collides with the unfinished transmission of other data frames after the occurrence of PLC between data frames. As a result of this collision, the ACK frame may be corrupted and the corresponding data frame needs to be retransmitted. We call this phenomenon "ACK Corruption" (AC). We identify the characteristic of AC via extensive experiments and simulations. Our study reveals that AC can occur in all IEEE 802.11 variants and its chance is dependent upon the relative signal strength between the stations and the MCS setting used. Further, we devise a way to avoid AC occurrence and evaluate its effectiveness.

1. Introduction

When two or more transmissions collide on the same channel in IEEE 802.11 network, usually all the transmissions fail. However, if the signal strength of a transmission is sufficiently higher than other transmissions, that transmission succeeds while others fail. This phenomenon is called *Physical Layer Capture* (PLC).

PLC in IEEE 802.11 has been extensively studied both analytically (e.g., [1–4]) and experimentally (e.g., [5–10]). In the existing studies, however, only the collision between data frames is considered.

In this paper, a new aspect of PLC, which we call *Acknowledgement Corruption* (AC), is investigated. AC occurs when a 802.11 ACK frame collides with data frame(s). So far, such collision has been studied only in the context of the hidden node problem. For example, in [11], the loss of ACK due to the collision with the transmission by the hidden nodes in the adjacent cells is studied. Note that "carrier sensing" of 802.11 MAC does not work when the hidden node problem exists. In [12], the loss of ACK due to the hidden node problem is studied in multihop wireless networks. In [13], the loss of data frames by the interference from ACK transmission in adjacent cells due to the hidden node problem is studied. In [14], ACK loss caused by non-802.11 devices that do not employ carrier sensing is studied.

In general, when there is no hidden node problem, ACK loss occurs very rarely since the 802.11 MAC guarantees exclusive transmission of ACK frame and ACK transmission is highly reliable as low-order MCS (Modulation Coding Scheme) is applied for ACK transmission, which guarantees robust transmission of ACK frame even at the presence of relatively severe noise.

In this paper, unlike the existing works, we consider a single WiFi cell with no hidden node. The target scenario of this paper is as follows. Suppose that there occur simultaneous transmissions of two (or more) data frames in the same WiFi cell. If the signal strengths of these two transmissions are sufficiently different, one data frame with stronger signal strength will be successfully delivered to its receiver thanks to PLC despite the collision. When the transmission of the stronger data frame is completed, the receiver will generate an ACK frame. At this point, it is possible that the other data frame, which has weaker signal strength, is still under transmission. Note that, unlike Ethernet, ongoing transmissions are not interrupted even if a collision occurs in 802.11. Now, the ACK frame for the stronger transmission will collide with the residual transmission of data frame of the weaker signal

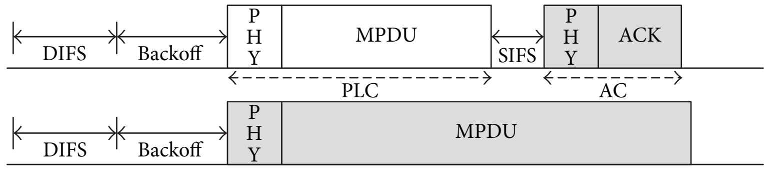


Figure 1: ACK Corruption in IEEE 802.11 b/g/a networks.

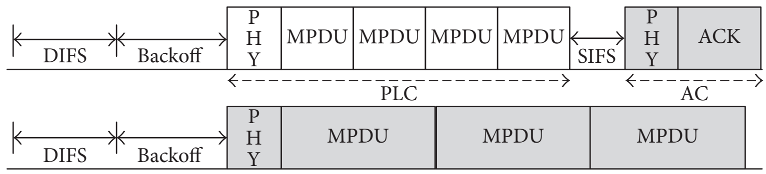


Figure 2: ACK Corruption in IEEE 802.11 n/ac networks.

strength. Depending on the relative signal strength of the ACK frame, the ACK frame may be corrupted as a result of the collision.

The occurrence of AC is conditioned on PLC (i.e., AC may occur only after the occurrence of PLC) and AC occurs during a very short duration. To assess the properties of AC, we perform carefully designed experiments and simulations. In our experiments, each 802.11 node is monitored by using a dedicated sniffer to catch all the transmissions from each node even if collision occurs. The sniffers which are placed right next to each 802.11 node record all the transmission attempts regardless of the transmission result (i.e., success or failure). We perform comprehensive experiments by varying the location settings and transmission bit rates. For large scale evaluation, we rely on simulations.

The results indicate that the chance of AC occurrences is dependent on many factors and is very high when certain conditions are met. We compare the properties of AC in 802.11a and 802.11n networks, which exhibit differences primarily due to the use of frame aggregation feature. Then we further investigate the properties of AC by focusing on how AC would affect the throughput of the 802.11 nodes. Based on the insight obtained from this investigation, we suggest an AC avoidance scheme to prevent (or reduce) AC occurrence and improve the overall throughput of the 802.11 network. Fairness among 802.11 nodes is also considered in this scheme.

The rest of this paper is organized as follows. In Section 2, we describe the necessary conditions for AC occurrence. In Section 3, we describe the experimental evaluation setting used in the paper and confirm the validity of the measurements. In Section 4, we present the results of 802.11a experiments. In Section 5, we present the results of 802.11n experiments, putting particular emphasis on the impact of frame aggregation on the occurrence of AC. In Section 6, we propose an AC avoidance scheme and evaluate its effectiveness. We conclude the paper in Section 7.

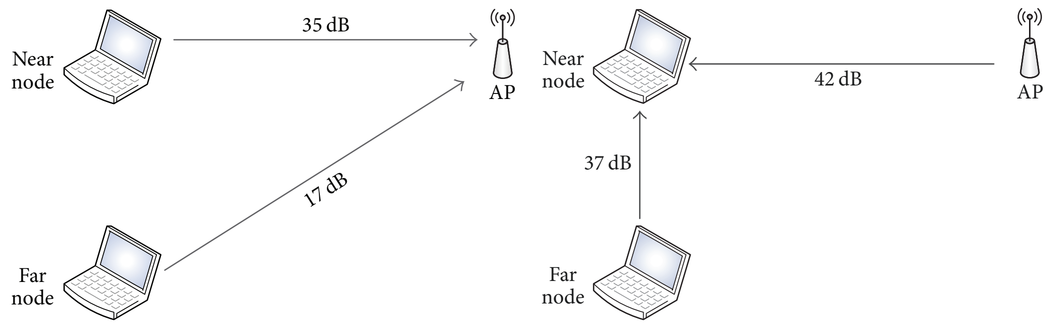
2. When Does "ACK Corruption (AC)" Occur?

In 802.11 MAC, ACK does not compete with regular data frames and therefore ACK does not normally collide with data frames. Barring the existence of the hidden node problem, the collision between ACK and a data frame occurs only after PLC between data frames occurs. AC occurs if the data frame is strong enough to corrupt ACK. The situation of AC occurrence in the 802.11 b/g/a networks is depicted in Figure 1. In Figure 2, the AC occurrence in the 802.11 n/ac networks where frame aggregation is applied is depicted. Only the collision involving two transmissions is considered in Figures 1 and 2 for simplicity, though the collision among more than two nodes is also possible.

Suppose that two user stations are connected to an 802.11 AP. One station is positioned nearer to the AP than the other station. We call the station that locates near the AP as *"Near node"* and the station locates far from the AP as *"Far node."* For AC occurrence, first of all, the data frame of *Near node* must be successfully received despite the collision with *Far node*, which means PLC. As a result, ACK is generated by the AP toward *Near node*. Meanwhile, the transmission of *Far node* fails and ACK will not be generated for it. Suppose that the *Far node*'s transmission lasts long enough, and this unfinished transmission collides with the ACK toward *Near node*. If the *Far node*'s transmission is strong enough as compared with the ACK, the ACK transmission will fail and the retransmission of *Near node* will be triggered.

Formally, there are three necessary conditions for the occurrence of AC. Firstly, PLC between data frames must occur and thus one of the data frames is successfully delivered to its receiver. Secondly, the other transmission should last long enough to cause collision with the ACK frame for the successful transmission. Thirdly, the signal strength of the failed data transmission should be strong enough to corrupt the ACK frame. If the signal strength of the ACK frame is sufficiently strong, another PLC that favors the ACK transmission will occur and ACK transmission will succeed (i.e., AC does not occur).

Throughout the paper, we use the following notations to represent the SNR relationship between nodes. $\mathrm{SNR}_{A\rightarrow B}$ denotes the SNR (Signal-to-Noise Ratio) of the transmission by the A, when measured at B. For instance, $\mathrm{SNR}_{\mathrm{Far}\rightarrow\mathrm{AP}}$ denotes the SNR (Signal-to-Noise Ratio) of the transmission by *Far node*, when measured at the AP. $(\mathrm{SNR}_{A\rightarrow C} - \mathrm{SNR}_{B\rightarrow C})$



(a) $\mathrm{SNR}_{\mathrm{Near}\rightarrow\mathrm{AP}} - \mathrm{SNR}_{\mathrm{Far}\rightarrow\mathrm{AP}} = 18\,\mathrm{dB}$ (b) $\mathrm{SNR}_{\mathrm{Far}\rightarrow\mathrm{Near}} - \mathrm{SNR}_{\mathrm{AP}\rightarrow\mathrm{Near}} = -5\,\mathrm{dB}$

FIGURE 3: Examples of SNR relationship notations.

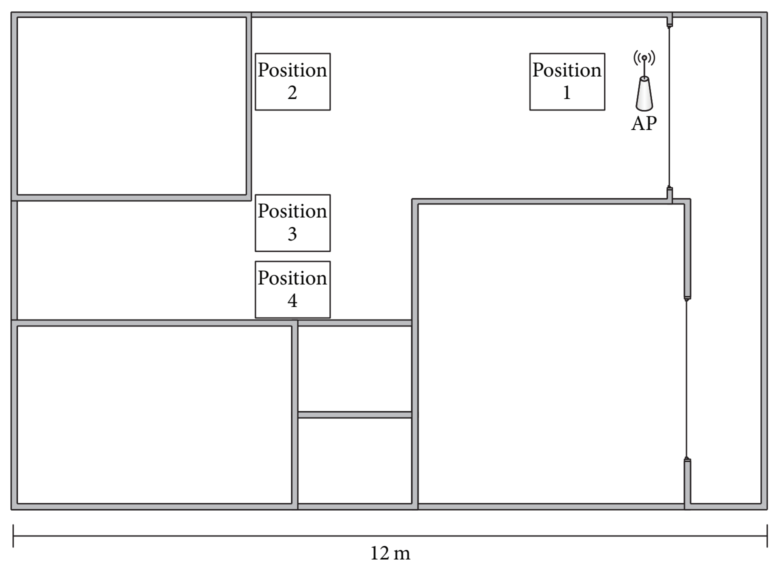


FIGURE 4: Experiment Testbed Layout.

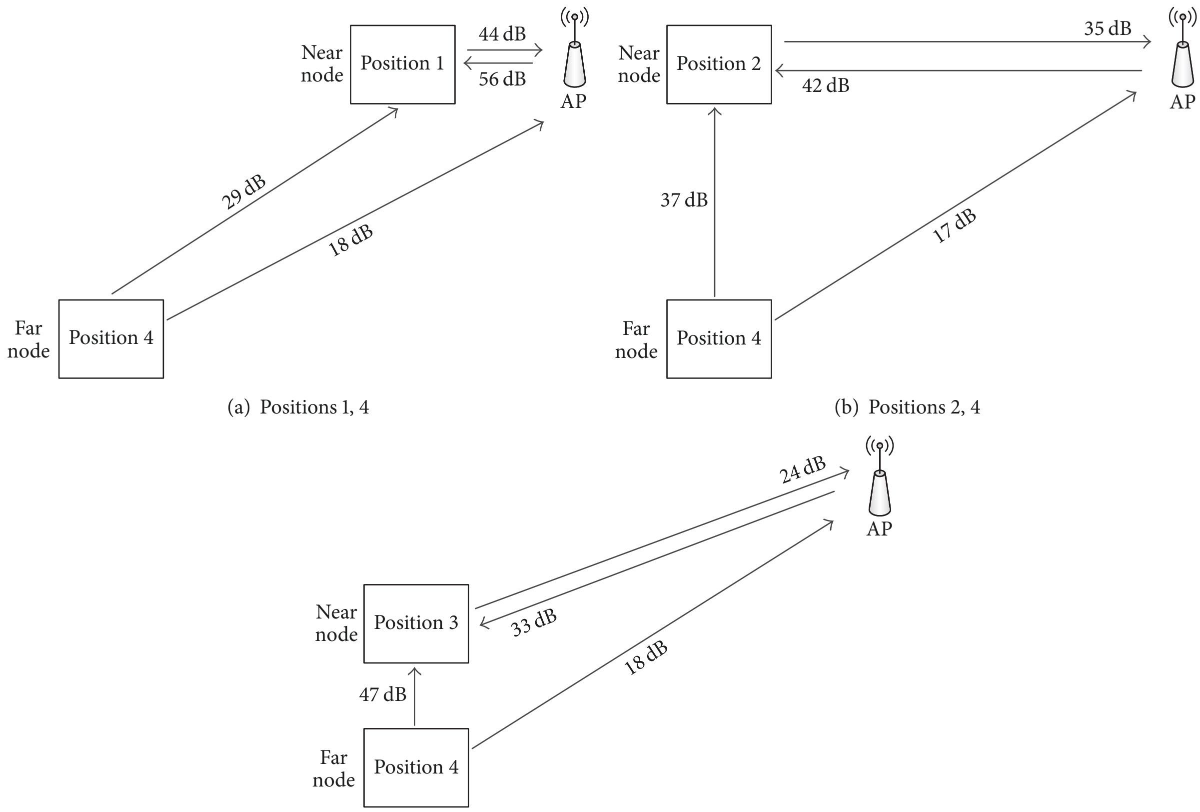
denotes the difference of SNR between A and B, which is measured at C. For instance, $(\mathrm{SNR}_{\mathrm{Far}\rightarrow\mathrm{Near}} - \mathrm{SNR}_{\mathrm{AP}\rightarrow\mathrm{Near}})$ denote the difference of SNR between AP and *Far node*, which are measured at *Near node*. Figure 3 illustrates some examples. In essence, if $(\mathrm{SNR}_{\mathrm{Near}\rightarrow\mathrm{AP}} - \mathrm{SNR}_{\mathrm{Far}\rightarrow\mathrm{AP}})$ is beyond a certain threshold, PLC occurs. If $(\mathrm{SNR}_{\mathrm{Far}\rightarrow\mathrm{Near}} - \mathrm{SNR}_{\mathrm{AP}\rightarrow\mathrm{Near}})$ is beyond a certain threshold, AC occurs.

3. Experiment Setting

3.1. Experiment Testbed. We use Linux laptops equipped with an Atheros AR9380-based PCI-E card that supports 802.11 b/g/a/n to build the experiment testbed. Ath9k [15] is used for the device driver. Three laptops are used. One serves as the AP and the other two functions serve as user stations. Figure 4 shows the layout of the experiment testbed. The position of the AP is fixed, while two user stations are located in four possible positions. Position 1 is closest to AP and Position 4 is farthest from AP. For the notational convenience, we denote by "Positions i, j" the case that *Near node* is at Position i and *Far node* is at Position j. By fixing the location of *Far node* at Position 4, we experiment three configurations: "Positions 1, 4," "Positions 2, 4," "Positions 3, 4." The SNR relationship of each case is shown in Figure 5 and Table 1.

Each laptop is equipped with an additional 802.11 interface card for the purpose of traffic sniffing. The sniffer interface catches all the transmissions and receptions at the PCI-E interface belonging to the corresponding laptop, and records the events in a log with timestamp. The resolution of the sniffer timestamp is one microsecond. Note that even when collision happens, the transmission of the collided packets can be detected by the sniffers associated with the sender stations (it is because PLC occurs at the sniffer interface of the sender station. Similar method has been used to catch collided packets as in [7]).

All the sniffing traces collected at each laptop are merged together into a single traffic trace. We synchronize the timestamps of the sniffing traces from different nodes by utilizing the beacons transmitted by the AP. The main challenge is that the software timestamp is unreliable due to the fluctuation of software timestamping delay. As a result, the timestamps of different nodes are not precisely aligned. We get around this issue by increasing the MAC slot time. With the extended slot



(a) Positions 1, 4

(b) Positions 2, 4

(c) Positions 3, 4

Figure 5: SNR relationships for experiments.

Table 1: SNR differences.

Near, Far node positions	Positions 1, 4	Positions 2, 4	Positions 3, 4
$SNR_{Near\rightarrow AP} - SNR_{Far\rightarrow AP}$	26 dB	18 dB	6 dB
$SNR_{Far\rightarrow Near} - SNR_{AP\rightarrow Near}$	−27 dB	−5 dB	14 dB

time, the time gap between data packets are also extended, so that it is possible to compensate the timestamp drift.

We fix the contention window size to 1 by setting both CW_{min} and CW_{max} to 1. As a result, only 0 or 1 slot is chosen for the random backoff value. This is solely to cause more collisions during the limited experiment duration. It does not affect the probability of PLC which is conditioned to the collision occurrence; that is, we only increase the chance of collision and the probability of PLC is unaffected. Iperf is used for traffic generation and the UDP packet size is set to 1536 bytes.

All the experiments are conducted on the 5.180 GHz frequency (i.e., channel 36) to avoid interference from non--802.11 devices. We confirmed that ACK loss by noise error or other external interferences is negligible before starting the experiments. Auto rate adaptation is disabled and the transmission bit rate is fixed to a certain selected value in the experiments. For the 802.11a experiments, we use four transmission bit rates which are 54 Mbps, 48 Mbps, 24 Mbps, and 12 Mbps. For the 802.11n experiments, 12 transmission bit rates are used. The MCS indices used in the 802.11n experiments are summarized in Table 2. Up to two spatial streams are used in the 802.11n experiments.

3.2. Detection of AC from Merged Traffic Trace. Table 3 shows an example of the merged traffic trace, which is obtained from the 802.11a experiments of "Positions 3, 4" setting. The transmission bit rate of *Near node* is set to 48 Mbps and that of *Far node* is set to 24 Mbps. Such information as packet length, transmission bit rate, sequence number, and retry flag are extracted from the 802.11 MAC frame header. The packet number at the first column is assigned for the convenience of packet identification.

The transmission duration of a packet is computed by accounting PLCP preamble length, PHY header length, and MAC frame length. If the transmission duration of two packets overlap more than a half of any packet transmission, we infer that a collision occurs. In Table 3, a collision occurs between Packet 27988 and Packet 27990 as their transmission durations overlap. Here, it can be observed that an ACK is generated by the AP for the Packet 27988, which indicates the occurrence of PLC between Packet 27988 and Packet 27990.

Table 2: MCS index used in the 802.11n experiments.

MCS index	Number of spatial streams	Modulation & coding	Transmission bit rate (Mbps)
1	1	QPSK 1/2	13
2	1	QPSK 3/4	19.5
3	1	16-QAM 1/2	26
4	1	16-QAM 3/4	39
6	1	64-QAM 3/4	58.5
7	1	64-QAM 5/6	65
8	2	BPSK 1/2	13
9	2	QPSK 1/2	26
11	2	16-QAM 1/2	52
12	2	16-QAM 3/4	78
13	2	64-QAM 2/3	104
14	2	64-QAM 3/4	117

Table 3: Example of merged traffic trace.

Packet number	Time (sec)	Sender	Receiver	Length (byte)	Sequence number	TX rate (Mbps)	Info	Retry flag
27986	62.227381	Near	AP	1536	437	48	Data	0
27987	62.227426	AP	Near	14		24	ACK	0
27988	62.232190	Near	AP	1536	438	48	Data	0
27989	62.232236	AP	Near	14		24	ACK	0
27990	62.232448	Far	AP	1536	3393	24	Data	0
27991	62.237000	Near	AP	1536	438	48	Data	1
27992	62.237045	AP	Near	14		24	ACK	0
27993	62.241809	Near	AP	1536	439	48	Data	0
27994	62.241854	AP	Near	14		24	ACK	0

Notice that the sequence number of Packet 27991 is the same as that of Packet 27988, which means that Packet 27991 is a retransmission. In other words, even if the AP sends an ACK to *Near node* for Packet 27988, *Near node* retransmits that packet. This happens because the ACK is corrupted due to the collision with the residual transmission of Packet 27990 which is send by *Far node*. We conclude that AC occurs in this case.

The probability of PLC and the probability of AC are computed as follows:

$$\begin{aligned} &\text{The probability of PLC} \\ &\quad = \frac{\text{the number of PLC occurrences}}{\text{the number of data frame collisions}}, \\ &\text{The probability of AC} \\ &\quad = \frac{\text{the number of AC occurrences}}{\text{the number of PLC occurrences}}. \end{aligned} \tag{1}$$

3.3. Validation of Experiment Setting. In this section, we examine the accuracy of our measurements. To this end, we compare the statistics extracted from the 802.11 device driver with the result obtained by analyzing the traffic trace. Specifically, we compare the number of nonretry data transmissions (i.e., the cases that the transmission does not suffer collision or channel error).

We examine the accuracy for all the experiment settings used in the paper. For each type of link (i.e., 802.11a and 802.11n), three configurations of node positions are tested, that is, "Positions 1, 4," "Positions 2, 4," and "Positions 3, 4." For each position configuration, various combinations of MCS settings are tested. The ACK frame is encoded by using 24 Mbps by default and is encoded in a lower MCS if the transmission bit rate of the data frame is lower than 24 Mbps.

The comparison results are given in Tables 4 and 5. The blank in the tables indicates that the corresponding MCS setting is infeasible due to too high channel loss rate and consequently is excluded from the experiments. Nearly 100% match is obtained in almost all cases, which confirms the accuracy of our experiment results.

4. 802.11a Experiment Results

We conduct experiments under the settings listed in Table 5. The probability of PLC and the probability of AC obtained from the experiments are plotted in Figures 6 and 7. Each setting is experimented three times and the average value obtained from the experiments is plotted in the results.

4.1. The Probability of PLC. In the case of "Positions 1, 4," the probability of PLC is nearly 100% regardless of the MCS setting used. It means that PLC almost always occurs when the transmissions of *Near node* and *Far node* collide. It is

Table 4: Comparison of the merged traffic trace and system-provided statistics (802.11n).

	MCS index		Positions 1, 4		Positions 2, 4		Positions 3, 4	
	Near	Far	Near	Far	Near	Far	Near	Far
802.11n (two spatial streams)	14	9	86.52%	99.87%	-	-	-	-
	13	9	99.80%	99.88%	-	-	-	-
	12	9	99.94%	99.90%	92.03%	99.76%	-	-
	11	9	99.83%	99.83%	99.68%	99.78%	96.75%	92.97%
	9	8	99.95%	99.81%	99.93%	99.76%	99.95%	99.13%
802.11n (one spatial stream)	7	3	99.64%	99.91%	-	-	-	-
	6	3	99.89%	99.89%	99.83%	99.87%	-	-
	4	2	99.98%	99.89%	99.96%	99.82%	99.94%	99.88%
	3	1	99.91%	99.81%	99.94%	99.8%	99.92%	99.87%
	3	3	99.95%	99.88%	99.96%	99.67%	99.87%	99.49%
	1	3	99.92%	99.90%	99.93%	99.81%	99.91%	99.89%

Table 5: Comparison of the merged traffic trace and system-provided statistics (802.11a).

	Bit rate		Positions 1, 4		Positions 2, 4		Positions 3, 4	
	Near	Far	Near	Far	Near	Far	Near	Far
802.11a	54 Mbps	24 Mbps	99.83%	99.57%	99.66%	99.59%	-	-
	48 Mbps	24 Mbps	99.84%	99.5%	99.72%	99.62%	99.41%	99.49%
	24 Mbps	24 Mbps	99.84%	99.95%	99.88%	99.97%	99.73%	99.57%
	12 Mbps	24 Mbps	99.63%	99.77%	99.9%	99.86%	99.34%	99.72%

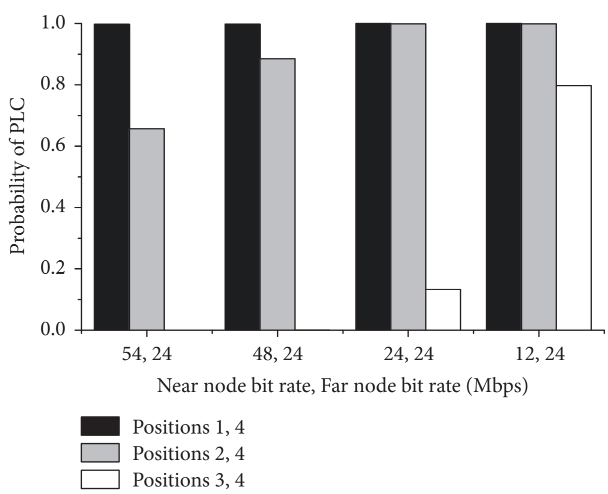


Figure 6: The probability of PLC in 802.11a experiments.

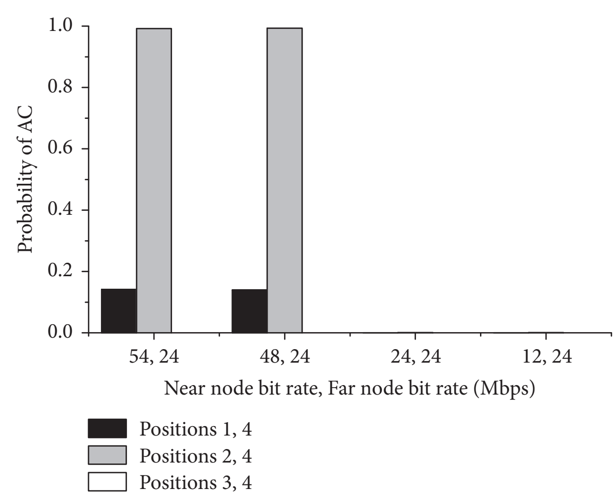


Figure 7: The probability of AC in 802.11a experiments.

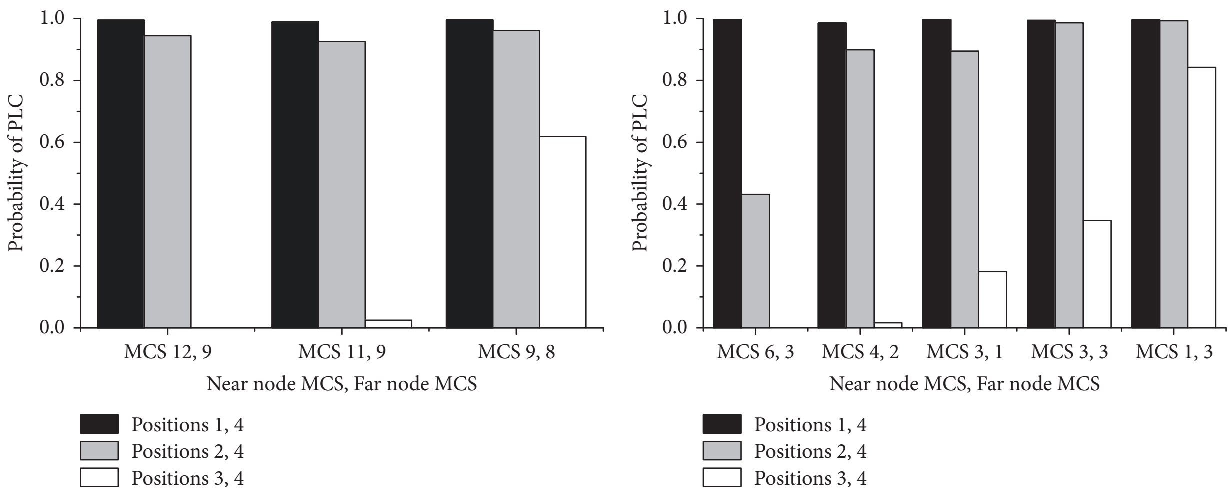
because the SNR difference between *Far node* and *Near node* to the AP is high; that is, ($SNR_{Near\to AP} - SNR_{Far\to AP}$) is high.

By contrast, in the case of "Positions 3, 4," the probability of PLC in the first two settings in Figure 6 is zero. PLC occurs only when *Near node* uses low-order MCS, that is, the last two settings. It is because the SNR difference between *Far node* and *Near node* to the AP is relatively small (i.e., ($SNR_{Near\to AP} - SNR_{Far\to AP}$) is low) and the transmission of *Near node* can survive only when more reliable MCS (i.e., low-order MCS) is used.

In the case of "Positions 2, 4," the MCS of *Near node* is the key factor as in the case of "Positions 3, 4." As more reliable MCS is used for *Near node*, the probability of PLC increases and the last two settings produce 100% PLC. As the transmission bit rate of *Near node* decreases, the probability of PLC increases.

Note that the MCS of *Far node* does not affect the probability of PLC. Only the relative location of *Far node* affects the probability of PLC.

4.2. The Probability of AC. Figure 7 shows the probability of AC obtained in the 802.11a experiments. First of all, regardless of the node positions, AC never occurs in the last two MCS settings (i.e., the transmission bit rates of "24,



(a) 2 spatial streams

(b) 1 spatial stream

Figure 8: The probability of PLC in 802.11n experiments.

24 Mbps" and "12, 24 Mbps" for *Near node* and *Far node*, resp.). It is because the transmission duration of *Near node* is equal or longer than that of *Far node*, which means that the ACK transmission from the AP to *Near node* never overlaps with the transmission of *Far node*. Recall that the packet size is set to be equal for both *Near node* and *Far node*.

Now let us consider only the first two MCS settings (i.e., the transmission bit rates of "54, 24 Mbps" and "48, 24 Mbps"). In the case of "Positions 3, 4," the probability of AC is zero since the probability of PLC is zero for this case. Recall that AC may occur only after PLC occurs.

In the case of "Positions 1, 4," the probability of AC is about 15%. Since the transmission duration of *Far node* is at least twice longer than that of *Near node*, the ACK transmission always overlaps with the transmission of *Far node*. In this case, however, the signal strength of ACK is strong enough to overpower the signal strength of *Far node* in most cases, that is, $SNR_{AP\rightarrow Near} > SNR_{Far\rightarrow Near}$. As a result, the chance of AC is relatively low.

If $SNR_{AP\rightarrow Near} < SNR_{Far\rightarrow Near}$, the probability of AC will be much higher. Indeed, we observe the probability of AC being 100% in the case of "Positions 2, 4."

5. 802.11n Experiment Results

5.1. The Probability of PLC. With respect to the probability of PLC, the 802.11n experiments produce very similar results to the 802.11a experiments. The results are plotted in Figure 8. Firstly, when ($SNR_{Near\rightarrow AP} - SNR_{Far\rightarrow AP}$) is high in such cases as "Positions 1, 4," the probability of PLC is nearly 100% regardless of the MCS setting used. When ($SNR_{Near\rightarrow AP} - SNR_{Far\rightarrow AP}$) is low, PLC rarely occurs unless *Near node* uses very reliable MCS (i.e., low-order MCS) such as MCS 1 for single stream, and MCS 9 for two streams. Secondly, PLC occurs more often when more reliable MCS (i.e., low-order MCS) is used for *Near node*, the same as in the 802.11a experiments.

5.2. The Probability of AC. Similar to the 802.11a experiments, AC rarely occurs in the case of "Positions 1, 4" and "Positions 3, 4." Therefore, we only show the results of the "Positions 2, 4" experiment in Figure 9. In this setting, $SNR_{Far\rightarrow Near}$ is high enough to cause AC as compared to $SNR_{AP\rightarrow Near}$.

The probability of AC in the 802.11n experiments is different from that of the 802.11a experiments in two aspects. Firstly, when *Near node* uses a higher-order MCS than *Far node*, the probability of AC is nearly 100% in the 802.11a experiments (see "Positions 2, 4" case in Figure 7). By contrast, the probability of AC is below 40% in the 802.11n experiments. Secondly, when *Near node* uses a lower or equal order MCS than *Far node*, the probability of AC is zero in the 802.11a experiments. In the 802.11n experiments, however, the probability of AC is 10~30%.

These differences are caused by the frame aggregation of 802.11n. Unlike 802.11a, multiple frames are aggregated into a single frame in 802.11n as depicted in Figure 10. As a result, the transmission duration of *Near node* and *Far node* is not necessarily determined by the MCS used. Even if one node uses higher-order MCS than another node, the transmission duration may be longer if more frames are aggregated. Consequently, the transmission of *Far node* may not overlap with the ACK frame for *Near node*, even if *Near node* uses higher-order MCS than *Far node*. On the flip side of the coin, the transmission of *Far node* may overlap with the ACK frame for *Near node*, even if *Near node* uses lower order MCS than *Far node*.

In Sections 5.3 and 5.4, we assess the impact of frame aggregation on AC in detail.

5.3. The Degree of Frame Aggregation in Real Applications. In the 802.11n standard, the maximum transmission duration of an AMPDU (Aggregated MAC PDU) is declared to 10 milliseconds, though it is an implementation dependent value (i.e., 4 milliseconds in Ath9k device driver). In practice, however, most AMPDU frames have far shorted transmission durations. It is because the 802.11n interface does not forward

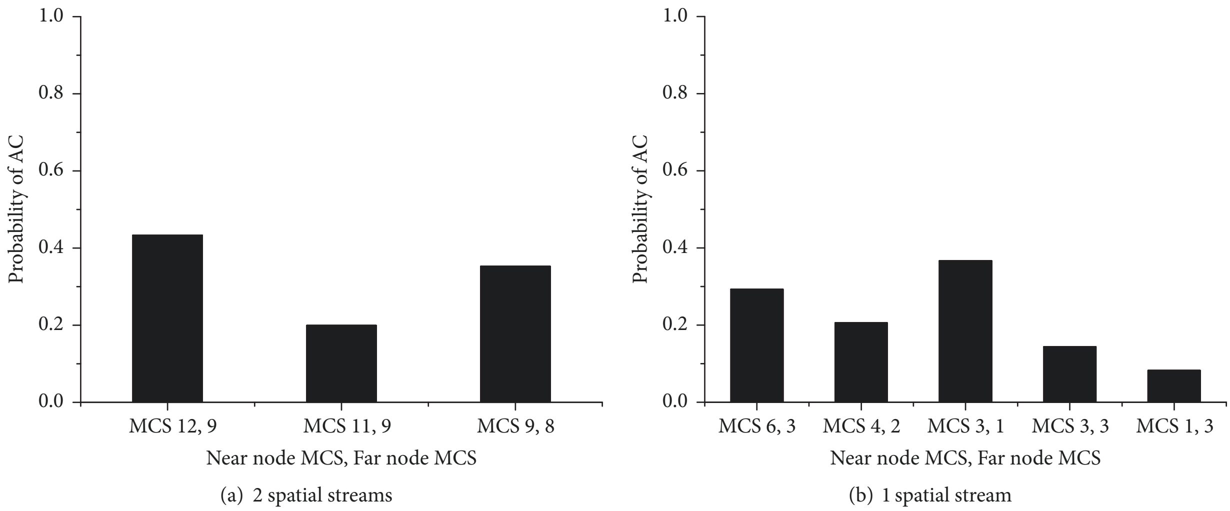


Figure 9: The probability of AC in 802.11n experiments: Positions 2, 4.

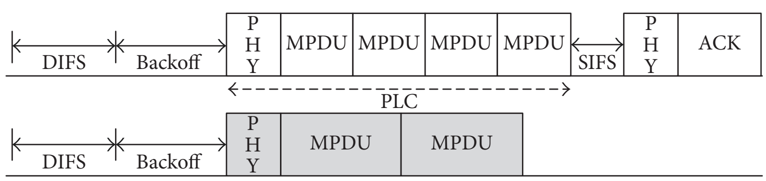


Figure 10: Frame aggregation and transmission duration.

the received frames to the upper layer until there is no hole in the "Block Acknowledgement (BA)." If there is any error in the MPDUs of the AMPDU, the lost MPDU must be retransmitted. As a result, MPDU frame aggregation is not fully utilized even if the sender has enough PDUs to transmit.

To analyze the degree of frame aggregation that is achieved in the real world, we experimentally measure the transmission duration of AMPDU frames of three types of applications: *Real time interactive game*, *File download*, and *Live streaming*. *Real time interactive game* is a TCP-based online web game that constantly sends the player's input to the server and receives the current status of the game from the server. *File download* is a TCP-based application that downloads a large file from a server. *Live streaming* is a UDP-based application that downloads a live video stream from a server. Experiments are conducted in a room and the user node is located 3 meters away from the AP.

The measurement results are plotted in Figure 11. It can be observed that most AMPDUs are far shorter than 4 milliseconds, while the degree of frame aggregation is highly dependent on the application used. *File download* produces much longer transmission durations than *Real time interactive game* and *Live streaming*. We also measure the degree of frame aggregation in an uncontrolled environment, that is, in a public cafe. The measurement result is plotted in Figure 11, marked by *Public place*.

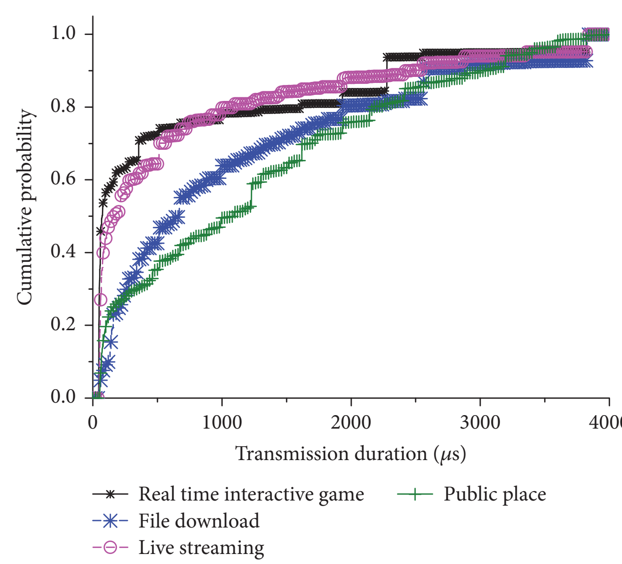
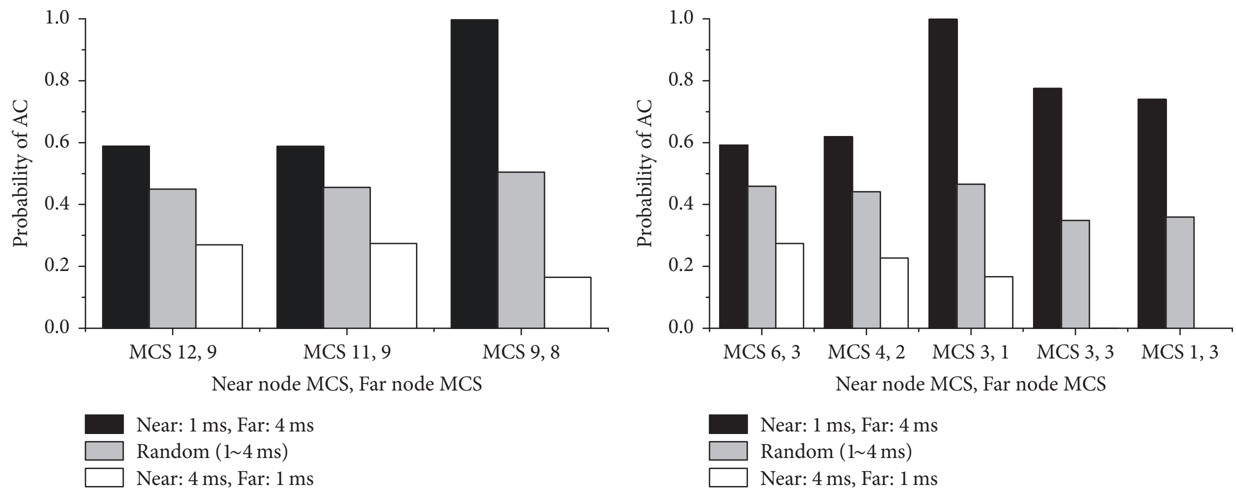


Figure 11: Distribution of transmission durations of AMPDU data frames from real traffic.

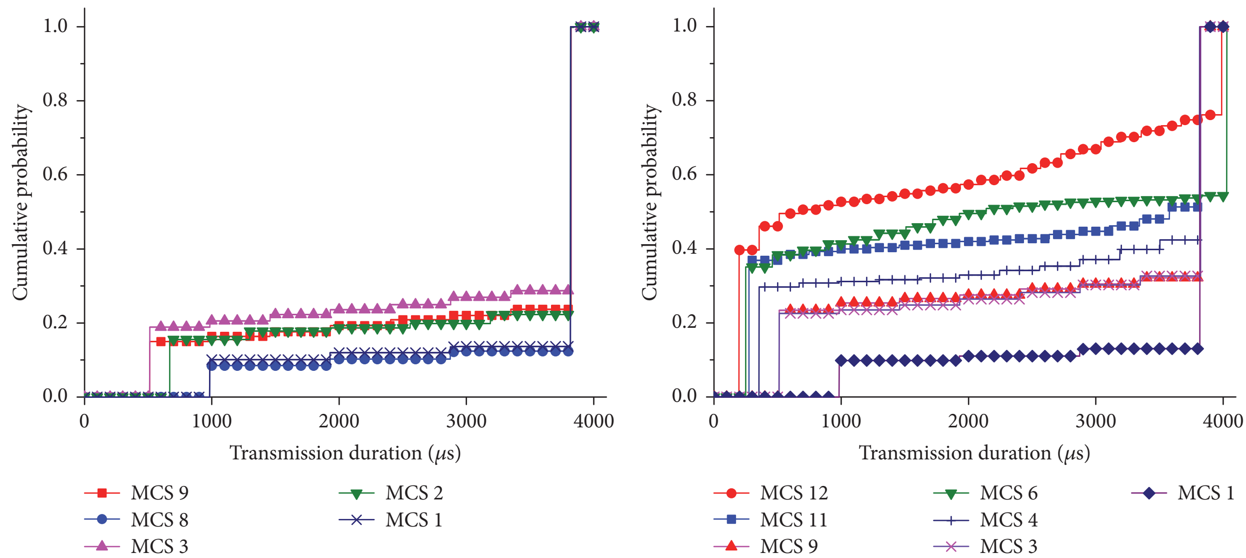
5.4. The Impact of Frame Aggregation on AC. We investigate the impact of frame aggregation on AC by intentionally limiting the degree of frame aggregation. To this end, the maximum transmission duration of an AMPDU is reduced to less than the default value of 4 milliseconds. Three configurations are tested: (i) "Near: 1 ms, Far: 4 ms" in which the maximum transmission duration of *Near node* is reduced to 1 millisecond, (ii) "Near: 4 ms, Far: 1 ms" in which the maximum transmission duration of *Far node* is reduced to 1 millisecond, and (iii) "Random (1~4 ms)" in which the



(a) 2 spatial streams

(b) 1 spatial stream

Figure 12: AC probability when the maximum AMPDU transmission duration is varied (“Positions 2, 4” setting is used).



(a) Far node in “Near: 1 ms, Far: 4 ms” experiment

(b) Near node in “Near: 4 ms, Far: 1 ms” experiment

Figure 13: Distribution of transmission durations of AMPDU frames (“Positions 2, 4” setting is used).

maximum transmission duration is randomly chosen from 1 to 4 milliseconds for both *Near node* and *Far node*.

The results of “Positions 2, 4” experiment are plotted in Figure 12. As expected, in general, the “Near: 1 ms, Far: 4 ms” experiments result in higher AC probability than the “Near: 4 ms, Far: 1 ms” experiments. It is because the chance that the transmission of *Far node* lasts after the completion of the transmission of *Near node* is higher in the former case than in the latter case, which leads to the collision between ACK and the *Far node* transmission. However, since the actual frame length may be shorter than 4 milliseconds even if the maximum transmission duration is set to 4 milliseconds, AC does occur in “Near: 4 ms, Far: 1 ms” experiments.

Let us examine the transmission duration of each case in more detail. Figure 13(a) shows the distribution of transmission duration of *Far node* in the “Near: 1 ms, Far: 4 ms” experiment. It can be observed that a large portion of frames have the transmission duration of near 4 ms, while in most of the remaining cases the transmission duration is below 1 ms. The size of the portion of transmission duration near 4 ms is dependent on the MCS used by *Far node*. When *Far node* uses low-order MCS such as MCS 1 or MCS 8, the portion of 4 ms frames is about 90%, while it is only about 75% in case of MCS 3.

The portion of transmission duration below 1 ms is virtually zero in case of MCS 1 or MCS 8. The reason is as follows. As low-order MCS is used, the chance of transmission error decreases, and consequently the chance of creating short frames for retransmission decreases. As a result, the AC probability is nearly 100% when *Far node* uses low-order

MCS such as MCS 1 or MCS 8 in the "Near: 1 ms, Far: 4 ms" experiments (see Figure 12). Recall that MCS 8 is slower (i.e., more reliable) than MCS 3 as shown in Table 2.

The interplay between MCS and transmission duration is observed also in the "Near: 4 ms, Far: 1 ms" experiment. Figure 13(b) shows the distribution of transmission duration of *Near node* in the "Near: 4 ms, Far: 1 ms" experiment. It can be observed that a large portion of transmissions by *Near node* has far shorter duration than 4 ms. Actually, a significant portion of transmissions have smaller than 1 ms duration when high-order MCS such as MCS 12 or MCS 6 is used for *Near node*. It is because the transmissions with high-order MCS tend to experience more errors and as a result short frames for retransmission are more frequently generated. For this reason, in Figure 12, we observed relatively high AC probability when *Near node* uses high-order MCS even in the "Near: 4 ms, Far: 1 ms" experiment.

5.5. Multiple Far Nodes. Due to the difficulty of scaling up the testbed, we rely on simulation to examine the case that multiple *Far nodes* exist. To this end, we use a simulator which is custom-made for our purpose instead of public simulators (e.g., ns-2), since PLC is not properly simulated in those simulators [7]. In our simulator, we implement the PLC occurrences by applying the probability of PLC that is experimentally obtained in our testbed. That is, for given locations of *Near node* and *Far node* and given MCS setting used, PLC is triggered at the probability measured in the experiment. For example, in case of "Positions 2, 4," the PLC probability is set to 43.2% for "MCS 6, 3 setting." The 802.11 MAC protocol is implemented by using typical parameters; for example, CW_{min} and CW_{max} are 15 and 1023, respectively. We verified the correctness of our simulator by comparing the collision probability and the throughput obtained from our simulator with the results of the analytical model presented in [16].

We simulate up to 10 user nodes. One node (i.e., *Near node*) is placed at Position 2 and the remaining nodes (i.e., *Far node*) are placed at Position 4. When there is only one *Far node*, it becomes the same as the "Positions 2, 4" experiment. The same MCS setting as in the "Positions 2, 4" experiment is used. The degree of MPDU frame aggregation is randomly chosen from the traffic trace of *Public place* shown in Figure 11.

The experiment result indicates that the chance of AC in the "Positions 2, 4" setting is nearly 100% if the *Far node* transmission is long enough to collide the ACK frame for *Near node*. Based on this observation, we trigger AC occurrence if a collision occurs between *Far node* transmission and ACK frame in the simulation. The transmission error caused by noise is ignored in the simulation.

Figure 14 shows the probability of AC as the number of *Far nodes* is increased from 1 to 9. In case of one *Far node*, the probability of AC is around 41~42%, which is consistent with the results of "Positions 2, 4" experiment (see the case of "Random" in Figure 12). We conduct 100 times of simulation for each setting.

As the number of *Far nodes* increases, the probability of AC also increases. It is because as the number of *Far nodes* increases, the number of *Far nodes* involved in each collision increases as well. We plot the average number of nodes involved in a collision (including *Far node*) in Figure 15. As the number of *Far nodes* involved in a collision increases, the chance of having *Far node* with long transmission duration increases.

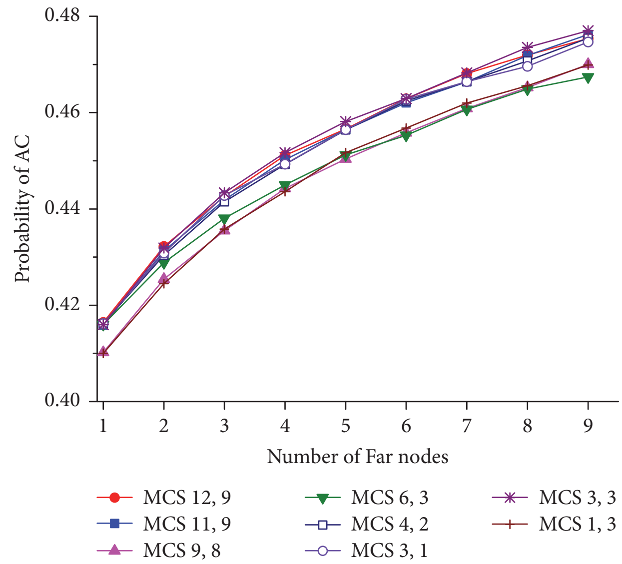


FIGURE 14: AC probability for different number of *Far nodes* (simulation result).

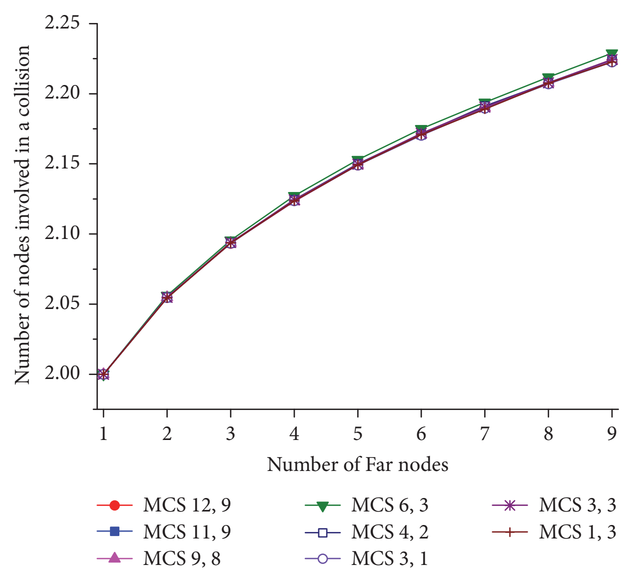


FIGURE 15: Average number of nodes involved in a collision (simulation result).

6. AC Avoidance Scheme

6.1. Throughput Decrease. In preceding sections, we show that AC occurs with high probability under certain conditions. The immediate consequence of AC is the nullification of PLC so that *Near node* with better channel condition has

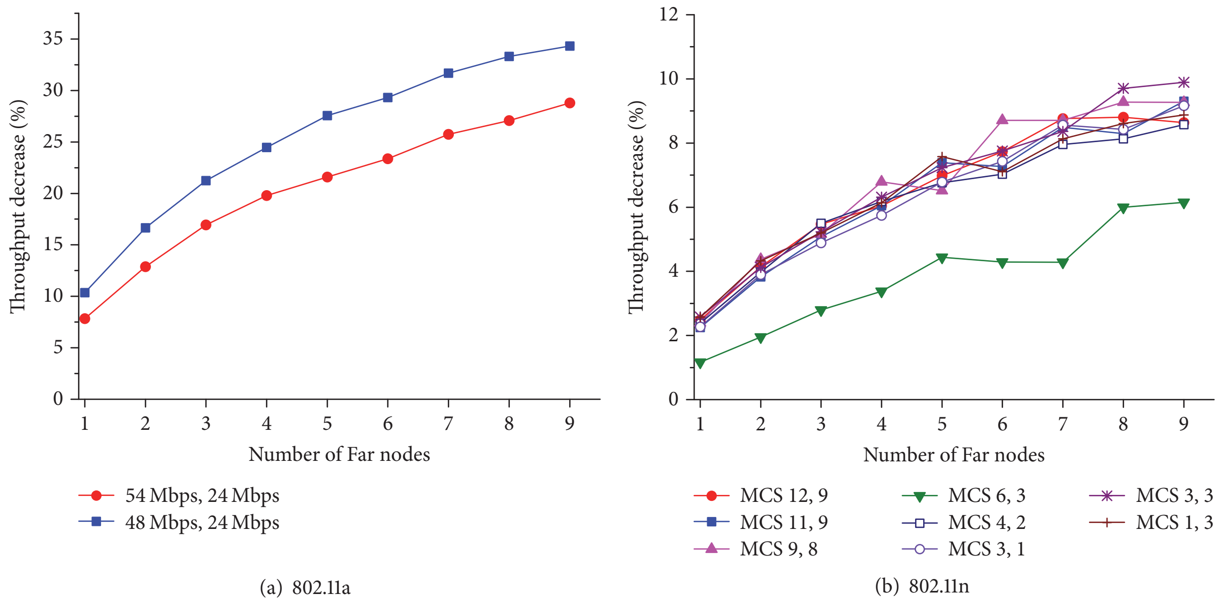


Figure 16: Throughput decrease of Near node due to AC.

to perform retransmission which is not necessary if AC does not occur. Note that PLC has a side effect of increasing the user throughput. In [10], a scheme is proposed to deliberately cause PLC to mitigate the impact of packet collisions (i.e., decrease of the total system throughput). From the view point of the user throughput, AC leads to "throughput decrease."

We assess how AC affects the throughput of *Near node* by comparing the results of AC-enabled simulation and AC-disabled simulation. In the real experiments, you cannot disable the occurrences of AC and therefore we rely on simulation for this analysis. In Figure 16, the simulation results are plotted as the number of *Far nodes* is varied. All the simulations are conducted in the "Positions 2, 4" setting. That is, *Near node* is placed at Position 2 and all the *Far nodes* are placed at Position 4. "Throughput decrease" is computed by ("Throughput of AC-disabled simulation" − "Throughput of AC-enabled simulation")/"Throughput of AC-enabled simulation."

The results of the 802.11a simulation are plotted in Figure 16(a). In the 802.11a simulation, frame aggregation is not used and all the data packets are 1536 bytes long for both *Near node* and *Far node*. Since the probability of PLC and the probability of AC in the "Positions 2, 4" setting are fairly high (See Figures 6 and 7), substantial throughput decrease is observed.

The results of the 802.11n simulation are plotted in Figure 16(b). In the 802.11n simulation, frame aggregation is used and the degree of MPDU frame aggregation is randomly chosen from the traffic trace of *Public place*. As compared to the results of the 802.11a simulation, the throughput decrease is smaller because of overall lower AC probability in the 802.11n networks. Notice that the throughput decrease of the case of "MCS 6, 3" is smaller than other cases. It is because the PLC probability of this case is particularly low (i.e., 43.2% as shown in Figure 8(b)).

6.2. AC Avoidance. We present a scheme to avoid (or reduce) AC occurrences. The key idea is to avoid the collision between the *Far node* transmission and the ACK transmission for *Near node*. To this end, there are two possible options: (i) extending the *Near node* transmission and (ii) shortening the *Far node* transmission. Suppose that all nodes have about the same amount of data to transmit. If we shorten the transmission duration of *Far node*, the system efficiency will be greatly reduced as *Far node* uses short frames with low-order MCS. Therefore, the first option makes more sense and we take this approach.

In the proposed scheme, *Near node* estimates *Minimum transmission duration* which is long enough to avoid AC. To compute this value, *Near node* firstly identifies the *Far nodes* that can potentially create AC. Recall that AC occurs only if ($SNR_{Far\to Near}$ and $SNR_{AP\to Near}$) is bigger than a certain threshold. By sniffing the ongoing transmissions, *Near node* obtains $SNR_{Far\to Near}$ and $SNR_{AP\to Near}$. The threshold is obtained experimentally. *Far nodes* that meet this condition are included in the candidate set.

Secondly, *Near node* checks if a *Far node* in the candidate set will produce PLC with the *Near node*. PLC occurs when ($SNR_{Near\to AP}$ − $SNR_{Far\to AP}$) is beyond a certain threshold. Otherwise, the *Far node* is excluded from the candidate set. This threshold is obtained experimentally, and there exist many previous works on measuring the PLC threshold. By assuming that the downlink channel and the uplink channel are symmetric, these values are indirectly estimated by using $SNR_{Far\to Near}$ and $SNR_{AP\to Near}$.

Finally, *Near node* computes the expected transmission duration of the *Far nodes* in the candidate set. The history of past transmissions by *Far nodes* within a certain time window is used. We consider two ways, which are "Moving average" and "Moving maximum." The former takes the average value in the time window, and the latter takes the maximum value

n_{near}: The node executing AC avoidance algorithm
N_{ac}: The set of the far nodes which can cause AC to n_{near}
N_{all}: The set of the nodes in the same WiFi cell as n_{near}
n_{near} monitors the channel to obtain the RSSIs and transmission durations of all the nodes in N_{all}
for each node $i \in N_{\text{all}}$ do
    If $(\text{SNR}_{i \to n_{\text{near}}} - \text{SNR}_{\text{AP} \to n_{\text{near}}}) >$ AC threshold then
        $N_{\text{ac}} \leftarrow N_{\text{ac}} \cup \{i\}$
    end if
end for
if $N_{\text{ac}} \neq \varnothing$ then
    for each node $j \in N_{\text{ac}}$ do
        if $(\text{SNR}_{n_{\text{near}} \to \text{AP}} - \text{SNR}_{j \to \text{AP}}) <$ PLC threshold then
            $N_{\text{ac}} \leftarrow N_{\text{ac}} - \{j\}$
        end if
    end for
end if
Minimum transmission duration $\leftarrow$ Moving avg. or moving max. of transmission durations of N_{ac}
Create data frame so that it is larger than *Minimum transmission duration*

Algorithm 1: AC avoidance algorithm.

in the time window. The *Minimum transmission duration* of *Near node* is decided so that the ACK transmission does not overlap with the expected *Far node* transmission.

Once a node computes its *Minimum transmission duration*, it tries to make the data frame size larger than is value. It is done by adjusting MCS in 802.11a and by adjusting the degree of frame aggregation in 802.11n. The proposed AC avoidance scheme is described in Algorithm 1.

6.3. Throughput Analysis of AC Avoidance. The effectiveness of our AC avoidance scheme is evaluated via simulation using the same setting as in Section 6.1. The result of 802.11a simulation (the case of "54 Mbps, 24 Mbps MCS setting") is presented in Figures 17 and 18. "Moving average" and "Moving maximum" produce identical results. It is because the packet size is fixed to 1536 byte for all nodes in this simulation. After applying the AC avoidance scheme, AC virtually disappears, and the throughput of *Near node* substantially increases.

One noteworthy observation is that when the number of *Far nodes* is one, the AC avoidance scheme rather decreases the throughput of *Near node*. It is because the penalty (i.e., throughput decrease) by downshifting the MCS for AC avoidance is bigger than the penalty caused by AC.

The result of 802.11n simulation (the case of "MCS 6, 3 setting") is presented in Figures 19 and 20. "Moving average" and "Moving maximum" produce different results because the degree of frame aggregation varies in the 802.11n simulation. As expected, "Moving maximum" eliminates the chance of AC while "Moving average" suffers occasional occurrences of AC. Unlike the 802.11a simulation, when the AC avoidance scheme is applied, *Near node* throughput is much higher than when it is disabled when the number of *Far nodes* is small. It is because the overheads for transmissions (i.e., SIFS, ACK reception, and PHY header) are reduced thanks to the high degree of frame aggregation triggered by the AC avoidance scheme so that the length of

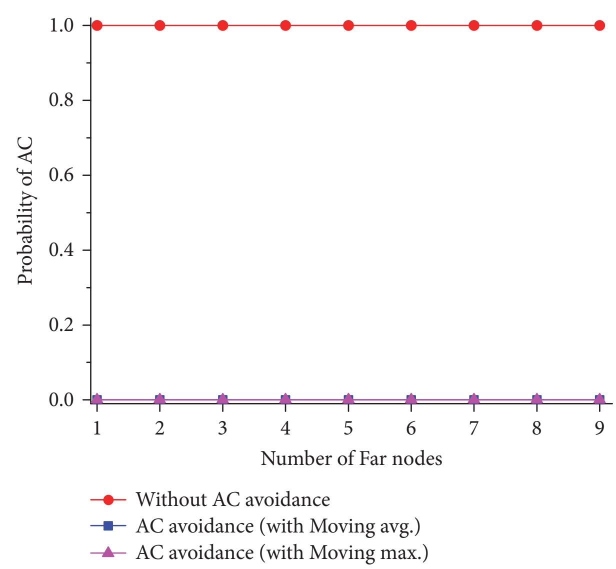


Figure 17: AC probability of *Near node* for the case of 54 Mbps, 24 Mbps (802.11a).

the aggregated frame is larger than *Minimum transmission duration*.

6.4. Fairness Analysis of AC Avoidance. As demonstrated in the previous section, our AC avoidance scheme increases the throughput of *Near node* by reducing AC. On the other hand, however, AC avoidance may affect fairness between *Near node* and *Far nodes*. To examine this aspect, we compare the fairness among nodes before and after applying our AC avoidance scheme. The result is plotted in Figure 21.

Jain's Fairness Index is used as a fairness metric and we measure the fairness of transmission opportunity (i.e., time used for successful transmission) instead of resultant throughput, because throughput is determined by the MCS

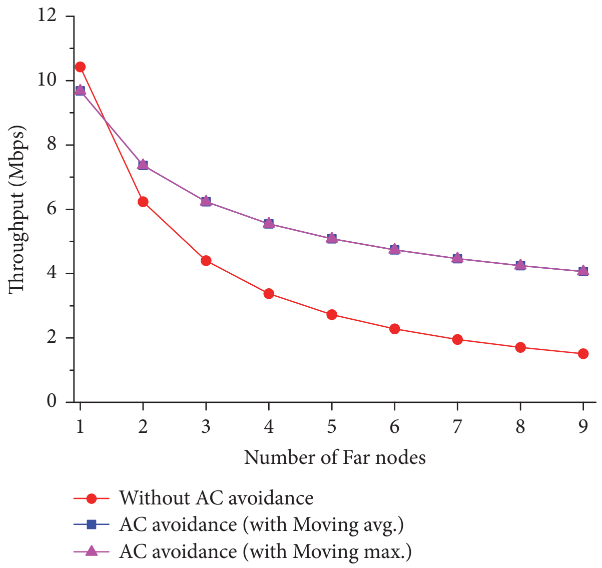


FIGURE 18: Throughput of *Near node* in the case of 54 Mbps, 24 Mbps (802.11a).

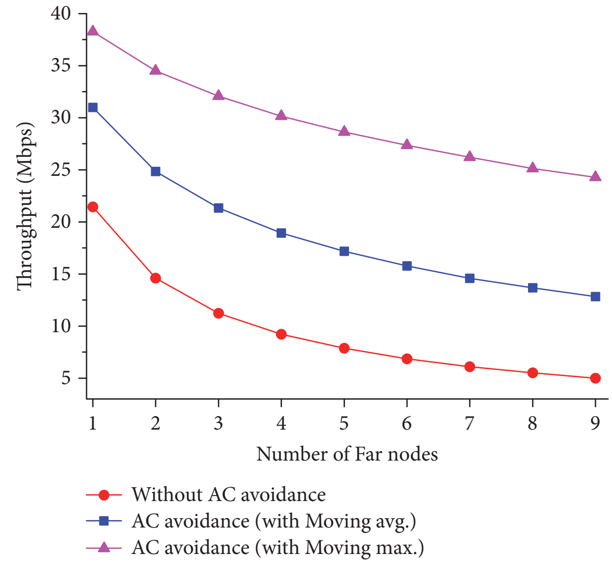


FIGURE 20: Throughput of *Near node* in the case of MCS 6, 3 (802.11n).

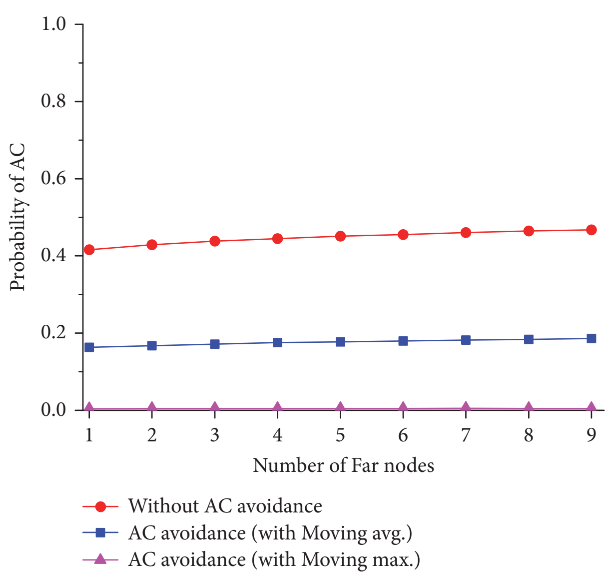


FIGURE 19: AC probability of *Near node* for the case of MCS 6, 3 (802.11n).

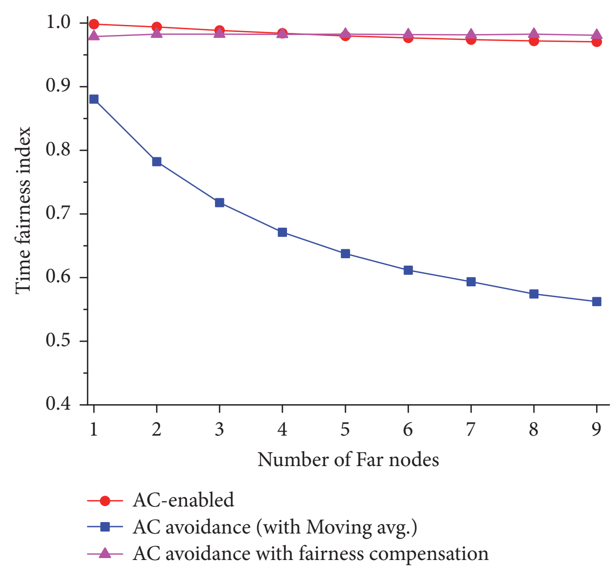


FIGURE 21: Time fairness in the case of MCS 6, 3 (802.11n).

being used and consequently it is harder to interpret. We call our comparison metric as "time fairness." Figure 21 reveals that the fairness is substantially worsen as the number of *Far nodes* increases after our AC avoidance scheme is applied, where fairness is relatively well maintained if AC avoidance is not applied. It is because PLC leads to transmission success for *Near node*, which does not increase its contention window size. Meanwhile PLC leads to transmission failure for *Far nodes*, which increases its contention window size.

In short, AC negates the effect of PLC while AC avoidance tries to sustain the effect of PLC. There are existing studies that exploit PLC to enhance the network performance such as [10, 17]. While AC avoidance scheme can introduce unfairness, it is beneficial to be applied to those schemes. For the situations where fairness needs to be maintained, we extend our AC avoidance scheme to mitigate unfairness as follows.

To reduce unfairness, the contention window size of *Near node* is increased when the AC avoidance scheme is applied. We call this extension by "fairness compensation." The effect of this extension is clearly shown in Figure 21. In the simulation, contention window is increased by the following equation:

Table 6: The probability of AC when auto rate adaptation is used.

	MCS index		Probability of AC
	Near	Far	
802.11n (two spatial streams)	12	9	43.31%
	11	9	19.93%
	9	8	35.25%
802.11n (one spatial stream)	6	3	29.31%
	4	2	20.58%
	3	1	36.67%
	3	3	14.36%
	1	3	8.23%
802.11n (auto)	Auto	Auto	9.52%

$$\mathrm{CW} = \min\left\{\mathrm{CW} \times \frac{\text{Minimum transmission duration}}{\text{Initial transmission duration}}, \mathrm{CW} \times \left(n\left(N_{\mathrm{ac}}\right)+1\right)\right\}. \quad (2)$$

CW is the contention window size currently set at Near node. *Initial transmission duration* is the transmission duration of *Near node* before applying the extension. *Minimum transmission duration* is the estimated transmission duration which is long enough to avoid AC. N_{ac} is the set of the nodes which can cause AC to *Near node*, and $n(N_{ac})$ is the number of these nodes. Different equations may be applied to make tradeoff between fairness and throughput. Figure 22 compares the aggregated throughput (i.e., the sum of the throughput of all nodes).

Note that this is different from not using AC avoidance at all. It is because if AC avoidance is disabled, AC occurs and the successful transmission by *Near node* is wasted. With the fairness extension, AC is avoided and the successful transmission by *Near node* is saved, and unnecessary decrease of bit rate at *Near node* is prevented. Unnecessary decrease of bit rate due to collision is a well-known issue as studied in [18].

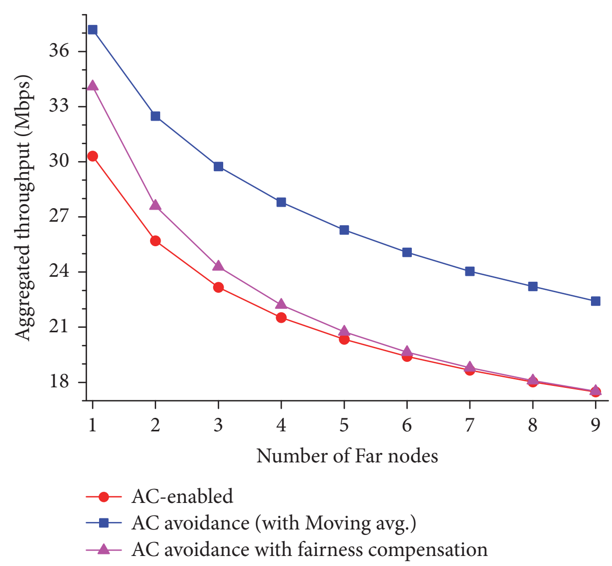


Figure 22: Aggregated Throughput in the case of MCS 6, 3 (802.11n).

6.5. Discussion on Auto Rate Adaptation. In our experiments reported in preceding sections, we disable the auto rate adaptation and set the transmission bit rate to a fixed value. When AC occurs, the *Near node*'s ACK is lost, which the *Near node* will consider a transmission failure. If auto rate adaptation is enabled, the transmission bit rate of the *Near node* will be lowered for the subsequent retransmission, which will extend the transmission duration of *Near node*. As a result of this action, the transmission duration of *Near node* may become longer than that of *Far nodes*, which breaks one of the necessary conditions for AC occurrence.

To examine the influence of auto rate adaptation on AC, we conduct the same experiments that are reported in Figure 9 by enabling auto rate adaptation for both *Near node* and *Far node*. The result of this experiment is summarized in Table 6. When auto rate adaptation is enabled, the probability of AC decreases to some degree, but we still observe substantial AC occurrences.

There are a number of reasons why AC still occurs even when auto rate adaptation is enabled. First of all, the frame aggregation mechanism of 802.11n can result in a longer transmission of a *Near node* even if the transmission bit rate of the *Near node* is equal to (or even lower than) that of the *Far node*. We have already explained the reason in detail. Secondly, after the bit rate of a *Near node* is lowered as a result of AC, the retransmission of the *Near node* will still experience AC if the bit rate is not low enough and the frame length of the *Near node* is still shorter than that of the far user. Only if the bit rate reduction is sufficient to prevent AC, the probability of AC will drop. However, as a result of successful transmission, the *Near node* will increase its bit rate again and therefore AC may occur again at the subsequent transmissions.

7. Conclusion

In this paper, we investigate a phenomenon that we call “ACK Corruption,” by which an ACK frame is corrupted due to the collision with unfinished transmission of other data frames. Our study reveals that AC can occur in all IEEE 802.11 variants. It is shown that AC occurrence is dependent upon the relative signal strength between the stations and the MCS setting used. AC directly affects (i.e., degrades) the throughput of each station and the total system throughput. In certain circumstance, the probability of AC occurrences is very high. We proposed a practical solution to avoid the AC occurrences.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean Government (MSIT) (no. 2016R1A2B4014505).

References

[1] Z. Hadzi-Velkov and B. Spasenovski, "Capture effect in IEEE 802.11 basic service area under influence of Rayleigh fading and near/far effect," in *Proceedings of the 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 2002*, pp. 172–176, Lisbon, Portugal, September 2002.

[2] H. Chang, V. Misra, and D. Rubenstein, "A general model and analysis of physical layer capture in 802.11 networks," in *Proceedings of the INFOCOM 2006: 25th IEEE International Conference on Computer Communications*, pp. 1–12, Barcelona, Spain, April 2006.

[3] F. Daneshgaran, M. Laddomada, F. Mesiti, M. Mondin, and M. Zanolo, "Saturation throughput analysis of IEEE 802.11 in the presence of non ideal transmission channel and capture effects," *IEEE Transactions on Communications*, vol. 56, no. 7, pp. 1178–1188, 2008.

[4] G. J. Sutton, R. P. Liu, X. Yang, and I. B. Collings, "Modelling capture effect for 802.11 DCF under Rayleigh fading," in *Proceedings of the 2010 IEEE International Conference on Communications, ICC 2010*, Cape Town, South Africa, May 2010.

[5] C. Ware, J. Judge, J. Chicharo, and E. Dutkiewicz, "Unfairness and capture behaviour in 802.11 adhoc networks," in *Proceedings of the IEEE International Conference on Communications (ICC '00)*, vol. 1, pp. 159–163, IEEE, New Orleans, La, USA, June 2000.

[6] J. Lee, W. Kim, S.-J. Lee et al., "An experimental study on the capture effect in 802.11a networks," in *Proceedings of the 2nd ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization, WiNTECH '07, Co-located with the ACM MobiCom 2007 Conference*, pp. 19–26, Montreal, Canada, September 2007.

[7] A. Kochut, A. Vasan, A. U. Shankar, and A. Agrawala, "Sniffing out the correct physical layer capture model in 802.11b," in *Proceedings of the 12th IEEE International Conference on Network Protocols, ICNP 2004*, pp. 252–261, Berlin, Germany, October 2004.

[8] S.-J. Han, T. Nandagopal, Y. Bejerano, and H.-G. Choi, "Analysis of spatial unfairness in wireless LANs," in *Proceedings of the 28th Conference on Computer Communications, IEEE INFOCOM 2009*, pp. 2043–2051, Rio de Janeiro, Brazil, April 2009.

[9] W. Wang, W. K. Leong, and B. Leong, "Potential pitfalls of the message in message mechanism in modern 802.11 networks," in *Proceedings of the 9th ACM MobiCom Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization, WiNTECH 2014*, pp. 41–48, Maui, Hawaii, USA, September 2014.

[10] P. Patras, H. Qi, and D. Malone, "Mitigating collisions through power-hopping to improve 802.11 performance," *Pervasive and Mobile Computing*, vol. 11, pp. 41–55, 2014.

[11] T. Bonald, A. Ibrahim, and J. Roberts, "Traffic capacity of multicell WLANS," in *Proceedings of ACM SIGMETRICS'08*, vol. 36, pp. 419–430, Annapolis, Md, USA, June 2008.

[12] J. Padhye, S. Agarwal, V. N. Padmanabhan, L. Qiu, A. Rao, and B. Zill, "Estimation of link interference in static multihop wireless networks," in *Proceedings of the the 5th ACM SIGCOMM conference*, p. 28, Berkeley, Calif, USA, October 2005.

[13] W. Wang, Q. Wang, W. K. Leong, B. Leong, and Y. Li, "Uncovering a Hidden wireless menace: Interference from 802.11x MAC acknowledgment frames," in *Proceedings of the 2014 11th Annual IEEE International Conference on Sensing, Communication, and Networking, SECON 2014*, pp. 117–125, Singapore, July 2014.

[14] R. Gummadi, D. Wetherall, B. Greenstein, and S. Seshan, "Understanding and mitigating the impact of RF interference on 802.11 networks," in *Proceedings of the ACM SIGCOMM'07*, vol. 37, pp. 385–396, Kyoto, Japan, August 2007.

[15] Ath9k, https://wireless.wiki.kernel.org/en/users/drivers/ath9k.

[16] Y. C. Tay and K. C. Chua, "A capacity analysis for the IEEE 802.11 MAC protocol," *Wireless Networks*, vol. 7, no. 2, pp. 159–171, 2001.

[17] P. Patras, H. Qi, and D. Malone, "Exploiting the capture effect to improve WLAN throughput," in *Proceedings of the 2012 IEEE Thirteenth International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, pp. 1–9, San Francisco, Calif, USA, June 2012.

[18] S. Rayanchu, A. Mishra, D. Agrawal, S. Saha, and S. Banerjee, "Diagnosing wireless packet losses in 802.11: separating collision from weak signal," in *Proceedings of the 27th IEEE Conference on Computer Communications (INFOCOM '08)*, pp. 1409–1417, Phoenix, Ariz, USA, April 2008.


Journal of Engineering
The Scientific World Journal
International Journal of Rotating Machinery
Journal of Sensors
Advances in Multimedia
Advances in Civil Engineering
Hindawi
Submit your manuscripts at
www.hindawi.com
Journal of Control Science and Engineering
Journal of Robotics
Journal of Electrical and Computer Engineering
Advances in OptoElectronics
VLSI Design
International Journal of Navigation and Observation
Modelling & Simulation in Engineering
International Journal of Aerospace Engineering
International Journal of Chemical Engineering
International Journal of Antennas and Propagation
Active and Passive Electronic Components
Shock and Vibration
Advances in Acoustics and Vibration