MoFA: Mobility-aware Frame Aggregation in Wi-Fi

Seongho Byeon, Kangjin Yoon
Okhwan Lee, Sunghyun Choi
Department of ECE and INMC
Seoul National Univeristy, Seoul, Korea
{shbyeon, kjyoon,
ohlee}@mwnl.snu.ac.kr
schoi@snu.ac.kr

Woonsun Cho, Seungseok Oh
CDD Team, Service Development Division
LG U+, Daejun, Korea
{vestsun, hanbawy}@lguplus.co.kr

ABSTRACT

IEEE 802.11n WLAN supports frame aggregation called aggregate MAC protocol data unit (A-MPDU) as a key MAC technology to achieve high throughput. While it has been generally accepted that aggregating more subframes results in higher throughput by reducing protocol overheads, our measurements reveal various situations where the use of long A-MPDU frames frequently leads to poor performance in time-varying environments. Especially, since mobility intensifies the time-varying nature of the wireless channel, the current method of channel estimation conducted only at the beginning of a frame reception is insufficient to ensure robust delivery of long A-MPDU frames. Based on extensive experiments, we develop MoFA, a standard-compliant mobility-aware A-MPDU length adaptation scheme with ease of implementation. Our prototype implementation in commercial 802.11n devices shows that MoFA achieves the throughput 1.8x higher than a fixed duration setting (i.e., 10 ms, the maximum frame duration according to IEEE 802.11n standard). To our best knowledge, this is the first effort to optimize the A-MPDU length for commercial 802.11n devices.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication

General Terms

Algorithms, Measurement, Performance

Keywords

IEEE 802.11 Wireless Local Area Network (WLAN); Frame Aggregation; Channel Estimation; Mobility

CoNEXT'14, December 2–5, 2014, Sydney, Australia.

ACM 978-1-4503-3279-8/14/12 ...$15.00.
http://dx.doi.org/10.1145/2674005.2674995.

1. INTRODUCTION

Wireless local area network (WLAN) technology has become an essential and indispensable part of our everyday life. Wi-Fi based on IEEE 802.11 WLAN standard has been one of the most successful wireless technologies supporting ever increasing demand of users. This tremendous success has led to significant growth of mobile WLAN data traffic volume primarily generated by pedestrian users. Accordingly, understanding the impact of mobility such as time-varying channel dynamics on the WLAN performance is becoming increasingly important.

IEEE 802.11 is evolving from 802.11a/b/g to 802.11n/ac in order to meet the much-needed high-throughput demand of smartphones, laptops, and tablet PCs [12]. To achieve high throughput, IEEE 802.11n defines two types of frame aggregation: MAC service data unit (MSDU) aggregation and MAC protocol data unit (MPDU) aggregation. In this paper, we deal with the latter, the aggregate MPDU (A-MPDU), which amortizes PHY protocol overhead over multiple frames by packing several MPDUs into a single A-MPDU. It is generally considered that A-MPDU is more efficient in error-prone environments thanks to the usage of block acknowledgements (BlockAcks) which allow each of the aggregated MPDUs (i.e., A-MPDU subframes) to be individually acknowledged and selectively retransmitted.

Understandingly, it has been believed that longer A-MPDU conveying more A-MPDU subframes always achieves higher throughput by reducing protocol overheads [8, 9, 11, 15]. All the existing studies have found the optimal length of MAC frames based on mathematical analysis and/or simulations, assuming a uniform distribution of errors across an entire A-MPDU.

However, our experimental results reveal strong evidence that the distribution of errors over the entire A-MPDU is not uniform, especially, for mobile users. For example, we observe that when long A-MPDU frames are used, the throughput is reduced by up to two thirds regardless of the channel condition at the receiver in time-varying channel environment, even if an appropriate PHY rate is selected. Furthermore, we find many scenarios where the performance actually degrades as the length of A-MPDU increases due to the limited channel compensation procedure executed by Wi-Fi devices. In such cases, the channel state information (CSI) measured using the physical layer convergence protocol (PLCP) preamble at the beginning of the A-MPDU may no longer be valid for subframes in the latter part of A-MPDU under the time-varying channel. Specifically, the

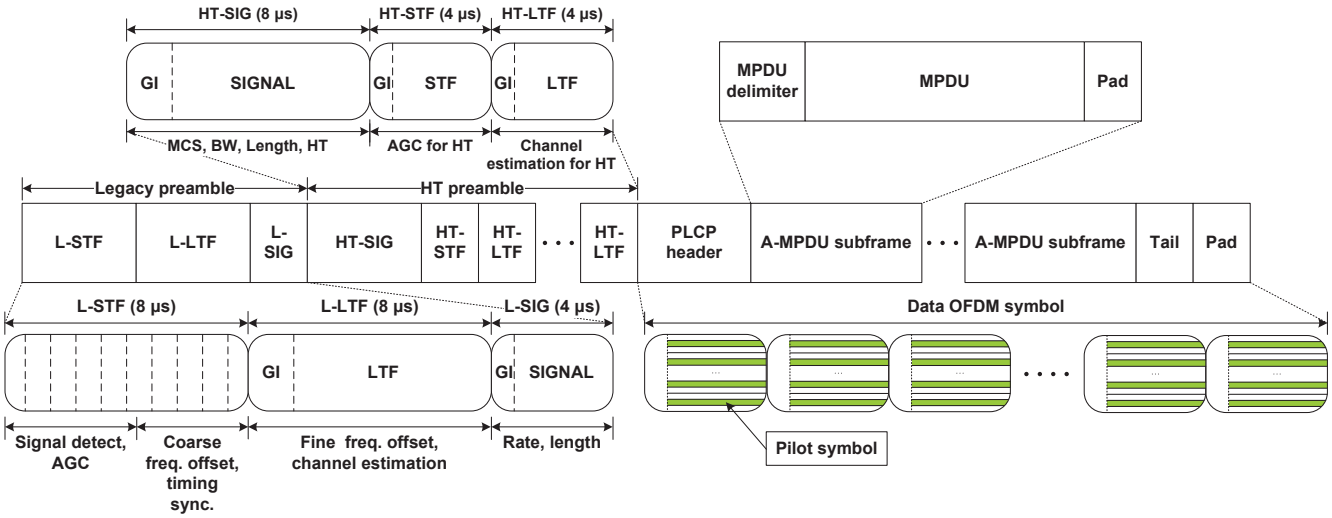


Figure 1: IEEE 802.11n MAC/PHY mixed-mode PPDU frame format of A-MPDU.

subframe error rate (SFER) may increase as the time gap between the preamble and subframe increases, since automatic gain control (AGC), timing acquisition, frequency acquisition, and channel estimation steps are conducted only during the PLCP preamble reception. It therefore leads to higher SFER in the latter part of A-MPDU than that in the beginning part when the channel condition substantially changes during the A-MPDU reception.

The primary contribution of this paper is a novel algorithm that dynamically adapts the length of A-MPDU that is robust against time-varying channels and without requiring any modification of the standard.

- We first analyze the wireless channel dynamics considering mobility in IEEE 802.11n WLAN through extensive measurements. From this, we reveal the fundamental problem of existing frame aggregation schemes manifested over a wide range of mobility and IEEE 802.11n PHY features.
- We then propose MoFA, a novel standard-compliant algorithm, which dynamically adapts the length of A-MPDU in real-time, by observing the mobility-caused channel dynamics in order to maximize the network performance.
- To demonstrate the effectiveness of MoFA, we conduct experiments using prototype implementation at the device driver of off-the-shelf 802.11n network interface cards (NICs).
- Finally, from the experimental results, the performance gain of our techniques over the standard 802.11n configuration is found to achieve up to 1.8x improvement.

To the best of our knowledge, this is the first work that develops mobility-aware adaptation of A-MPDU length that is robust against time-varying channels caused by mobility. We expect MoFA to enhance the performance of low error tolerant real-time applications such as online gaming and video streaming on a mobile device.

The rest of the paper is organized as follows. Section 2 introduces the background and describes our experimental setting. In Section 3, we empirically study the impact of mobility on IEEE 802.11n systems. Then, the detailed design of MoFA is presented in Section 4, and its implementation and our experimental results are presented in Section 5. Section 6 provides the related work, and finally, Section 7 concludes the paper.

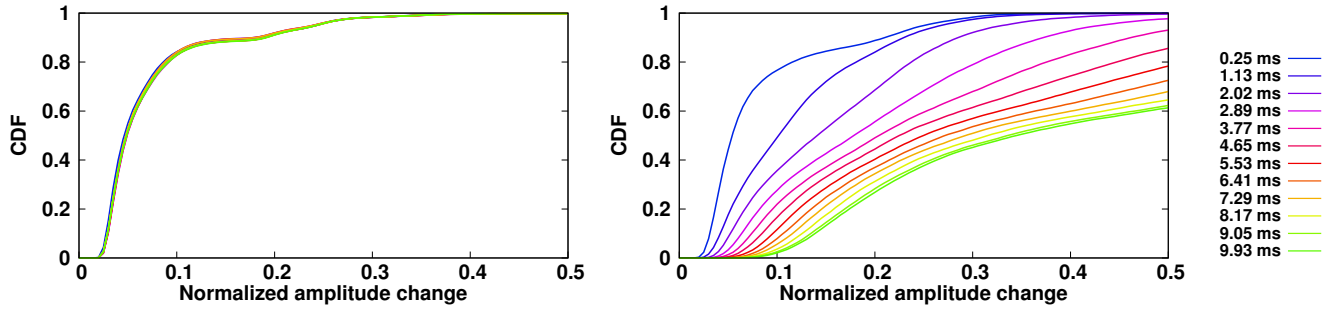
2. PREREQUISITES

In this section, we first present a brief description of the channel estimation method defined in the 802.11n standard. It is then followed by the 802.11n High Throughput (HT) features, and finally the experimental setting used for our measurements is presented.

2.1 Channel Estimation and Compensation

A received signal through wireless channel is distorted by fading, shadowing, and noise, dramatically changing in mobile environments. For this reason, an 802.11 receiver estimates the channel conditions to obtain the CSI by using training symbols in the PLCP preamble, located at the beginning of a PHY layer Protocol Data Unit (PPDU). As shown in Figure 1, the 802.11n PLCP preamble is composed of the legacy preamble (from IEEE 802.11a) and the HT preamble. The legacy preamble includes legacy short training field (L-STF) and legacy long training field (L-LTF), which are followed by a signal field (L-SIG). The L-STF is used by a receiver for signal detection, AGC, timing synchronization, and coarse frequency offset estimation. On the other hand, the L-LTF provides a means for the receiver to estimate the channel and fine frequency offset. Then, the HT-LTFs as part of the HT preamble are used to measure the CSI of multiple-input multiple-output (MIMO) channel [12].

A received signal can also be distorted by the symbol time offset (STO), caused by the difference of the local oscillators in the transmitter and receiver. This clock mismatch results in a phase rotation linearly proportional to the subcarrier index in frequency domain, which causes loss of orthogonality between subcarriers, and hence, pilot subcarriers are embedded in each OFDM data symbol to assist channel com-



(a) Static trace. (b) Mobile trace.

Figure 2: CDF of the normalized amplitude changes with varying time gap, τ.

pensation during the frame reception. In the 802.11n, 4 out of 56 subcarriers are dedicated to pilot subcarriers, eliminating the channel distortion from STO. With the support of pilot subcarriers, the estimated channel using the CSI measured from LTFs is interpolated to enable the coherent detection [16].

In summary, the channel is estimated using the PLCP preamble at the beginning of a frame reception. The receiver then compensates the OFDM data symbols according to the measured CSI from LTF assisted by pilot subcarriers. However, if the channel is dynamically changing during a frame reception, the receiver has no way to catch up with the CSI variation. Accordingly, when the transmission time of a frame is long, the current channel estimation process of the 802.11 might not be sufficient to decode the latter part of the frame, especially, in mobile environments.

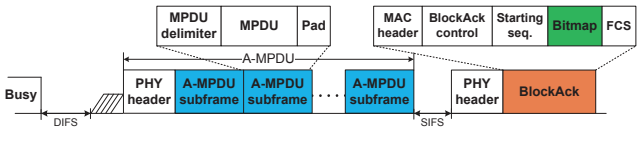


Figure 3: A-MPDU and BlockAck exchange.

2.2 IEEE 802.11n HT Features

2.2.1 *Frame aggregation*

Frame aggregation is a core feature of IEEE 802.11n to send multiple frames in a single transmission. Every data frame has a significant amount of overhead including PLCP preamble, MAC/PHY header, acknowledgement (ACK) transmission, and several Inter Frame Spaces (IFSs). These overheads often consume bandwidth comparable with the actual data payload, which results in much lower throughput compared with the underlying PHY data rate. To address this issue, the 802.11n defines two types of frame aggregation: MSDU and MPDU aggregation [12].

A-MSDU is literally the aggregation of MSDUs into a single frame at the upper MAC layer with the maximum size of 7,935 bytes. Since all aggregated MSDUs share a single MAC header and a single cyclic redundancy check (CRC) code, used for error detection by the receiver, the transmission of an A-MSDU fails as a whole even when just one of the aggregated MSDUs is corrupted.

On the other hand, A-MPDU, which is more commonly used in practice due to the merit explained below, combines multiple MPDUs into a single PPDU as shown in Figures 1 and 3. The maximum A-MPDU length is 65, 535 bytes, and the A-MPDU transmission time should be smaller than the maximum PPDU duration, *aPPDUMaxTime*, defined as 10 ms [12]. Unlike A-MSDU, A-MPDU is more efficient in high error rate environment, because all individual subframes (or MPDUs conveyed in subframes) are positively/negatively acknowledged using BlockAcks as shown in Figure 3, and hence, can be individually retransmitted.

2.2.2 *MCSs*

Modulation and coding schemes (MCSs) are used to denote the combination of the number of spatial streams, modulation, and code rate. A higher order MCS index indicates a higher PHY data rate provided by higher order modulation and/or higher code rate. It is generally known that lower order MCS is more robust than higher order MCS, while higher order MCS can lead to higher throughput as long as the channel condition is good. IEEE 802.11n basically supports 32 MCSs from MCS 0 (one stream using BPSK and 1/2-rate code) to MCS 31 (four streams using 64-QAM and 5/6-rate code).

2.2.3 *MIMO, SM, STBC and channel bonding*

IEEE 802.11n supports MIMO operation which utilizes multiple antennas at both transmitter and receiver. There are two transmission techniques which use multiple antennas, namely spatial multiplexing (SM) and space-time block coding (STBC). SM transmits multiple data streams through multiple antennas to enhance the data rate. On the other hand, STBC transmits multiple coded copies of a data stream through multiple antennas to improve the reliability. The 802.11n also supports channel bonding which uses 40 MHz bandwidth by combining two 20 MHz channels. Apparently, such bandwidth expansion achieves higher data rate.

2.3 Experimental Setting

We have conducted our experiments in a controlled office environment, i.e., the basement of our building. Figure 4 illustrates the floor plan used for our experiments, and points P1 to P9 represent different locations where stations are located. IEEE 802.11n devices supporting A-MPDU, 3x3 MIMO (SM and STBC), and 40 MHz channel bonding are used. The operating channel number 44, i.e., 5,220 MHz center frequency, is used, where no external interference has been detected. We mainly use a programmable 802.11n device, Qualcomm Atheros AR9380 NIC, along with *hostAP* [5]

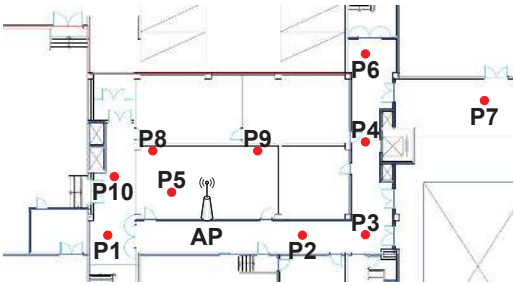


Figure 4: Experiment topology.

to build an AP on a *Ubuntu* 10.04 machine. Both AR9380 and IWL5300 (from Intel) NIC are utilized on the station side to consider various Wi-Fi devices [1]. We generate UDP downlink traffic from AP to stations using *Iperf* 2.0.5 to generate fixed length frames (of 1,534 bytes including MAC header) [3], and control the frame aggregation time bound by modifying the device drivers *ath9k* and *iwlwifi* for AR9380 and IWL5300, respectively [1, 2].

3. CASE STUDY

In this section, we first analyze the wireless channel dynamics based on the measurement results. Then, we investigate the impact of mobility on the performance of A-MPDU.

3.1 Temporal Selectivity

To illustrate our findings, we first present the temporal selectivity of the wireless channel by using CSI statistics in static and mobile environments. For the mobile scenario, the station comes and goes between P1 and P2 at an average speed of 1 m/s. We collect CSI traces from IWL5300 NIC using the modified device driver at the receiver side [4]. Specifically, the NIC reports measured 40,000 CSIs from HT-LTFs for 30 subcarrier groups in the form of 1×3 matrix [12]. We enable one antenna at a sender and three antennas at a receiver. The sender broadcasts a NULL data frame with MCS 0 using 15 dBm transmit power every 250 μs.

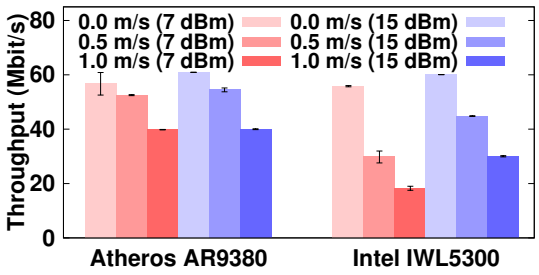
We employ the following metric representing the normalized amplitude changes to evaluate the temporal selectivity [7]:

$$\frac{\|A(t) - A(t+\tau)\|_2}{\|A(t+\tau)\|_2}, \tag{1}$$

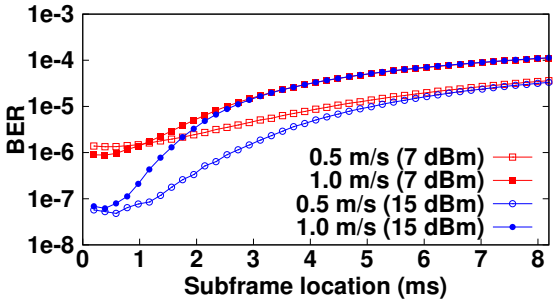
where the vector $A(t)$ represents all the subcarriers' amplitudes of the frame received at t, and τ is the time gap. $\|\cdot\|_2$ is the l^2-norm, and τ varies from 250 μs to 10 ms, which is *aPPDUMaxTime*.

Figure 2 shows the cumulative distribution function (CDF) of the normalized amplitude changes between CSI of two frames with varying τ. As shown in Figure 2(a), the amplitude variation remains steady in the static scenario. More than 85% of samples show the amplitude changes under 10% even for $\tau = 10$ ms.

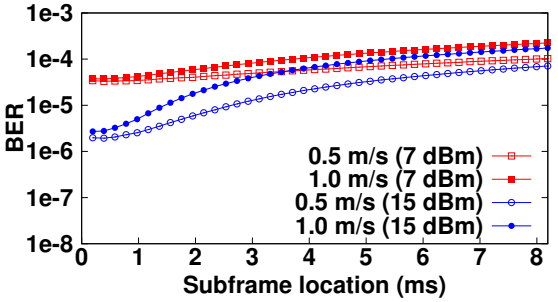
However, the amplitude variation increases with τ in the mobile scenario as shown in Figure 2(b). For $\tau = 10$ ms, the amplitude varies by more than 10% between two consecutive frames for over 95% of samples. Furthermore, for over 55% of samples, the amplitude changes by more than 30%.



(a) Throughput results.



(b) BER of AR9380 NIC.



(c) BER of IWL5300 NIC.

Figure 5: Impact of mobility.

Consequently, the error rate of the latter part of A-MPDU is expected to increase because the channel estimation is performed only during the PLCP preamble reception as explained in Section 2.1.

The observation from Figure 2 is highly related to the channel coherence time, which represents the time duration over which the wireless channel does not seem to be varying. Mathematically, it is defined as the time range over which the correlation coefficient of signal amplitudes is less than a threshold, i.e., 0.9, between two different signals [17] as follows:

$$\frac{\langle a_t a_{t+\tau}\rangle - \langle a_t\rangle\langle a_{t+\tau}\rangle}{\sqrt{\left[\langle {a_t}^2\rangle - \langle a_t\rangle^2\right]\left[\langle {a_{t+\tau}}^2\rangle - \langle a_{t+\tau}\rangle^2\right]}} \geq 0.9, \tag{2}$$

where a_t and $a_{t+\tau}$ represent the amplitudes of signals at time t and $t+\tau$, respectively, and $\langle x\rangle$ denotes the ensemble average of x. Using the above traces and Equation (2), the measured coherence time for the scenario of 1 m/s average speed is about 3 ms, which is much shorter than *aPPDUMaxTime*, i.e., 10 ms. It provides an important intuition

Table 1: Throughput with different time bounds

Time bound (μs)		0	1,024	2,048	4,096	6,144	8,192
Avg. # of aggregated frames		1	5	10	21	31	42
Throughput (Mbit/s)	0 m/s	31.19	50.95	56.47	59.62	60.50	60.90
	1 m/s	31.14	50.82	55.79	53.58	46.39	40.25
SFER (%)	1 m/s	0.05	0.13	1.10	10.01	23.33	34.00

that using A-MPDU without considering the wireless channel dynamics could severely deteriorate the throughput.

3.2 Impact of Mobility

We now examine how the limitation of the channel compensation procedure affects the reception of A-MPDU when mobility intensifies the time-varying nature of the wireless channel. As depicted in Section 2.3, we install a stationary AP equipped with AR9380, and use both AR9380 and IWL5300 on the station side. Saturated downlink traffic is sent from AP to a station which holds a position at P1 for the static scenario. For the mobile scenario, the station comes and goes between P1 and P2 at an average speed of 0.5 or 1 m/s. Frames are aggregated into A-MPDU frames, which are transmitted at the fixed MCS 7 (65 Mbit/s), employing 64-QAM and 5/6-rate code. The size of each subframe is set to 1,538 bytes including MPDU delimiter and padding bits, and hence, an A-MPDU can contain at most 42 subframes. Therefore, the actual transmission duration of a single A-MPDU is about 8 ms. The results are averaged over 5 runs, where each run lasts for 120 seconds.

We first investigate whether there exists significant performance degradation in mobile scenarios. As shown in Figure 5(a), we observe that the throughput decreases regardless of the NIC type and transmit power, as the degree of the mobility increases. Note that the wireless channel condition between AP and the station is pretty good in terms of signal-to-noise ratio (SNR). As an evidence, the results of the static case (0 m/s) achieve almost maximum throughput. However, in the case of the mobile scenario, the losses of throughput for AR9380 and IWL5300 are as high as one third and two thirds, respectively.

To investigate this further, we next analyze the bit error rate (BER) performance. Figure 5(b) shows the BER of A-MPDU subframes using AR9380 NIC on a logarithmic scale. The subframe location stands for the transmission starting instant of the corresponding subframe relative to the beginning of the PPDU over the air. We observe that subframes in the latter part of A-MPDU experience higher BER, and the curves show steep slope. The gradient of BER curves becomes higher as the average speed of the station increases. Additionally, BER curves converge depending on the mobility for both transmit powers (7 dBm and 15 dBm) regardless of the BER at the beginning of A-MPDU, since the mobility becomes the main cause of MPDU losses in the latter part of A-MPDU.

In other word, even if the appropriate PHY rate is used for transmission, this undesirable phenomenon can occur. Therefore, we can conclude that the adequate A-MPDU length selection is important to enhance the performance of A-MPDU. As shown in Figure 5(c), the result of IWL5300 shows a similar tendency even if RF circuit structure, antenna gain, and receiver sensitivity of these two NICs are different.

We calculate the optimal length of A-MPDU exhaustively based on the measured BER of AR9380 NIC with transmit

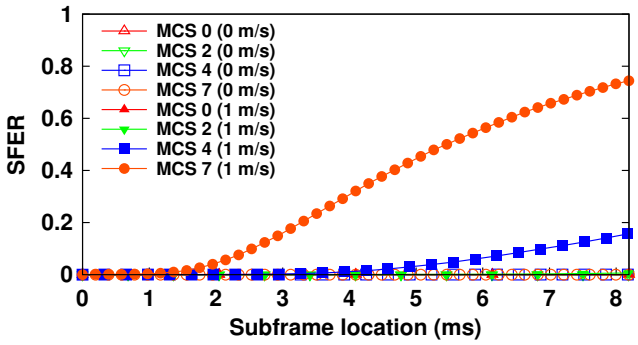


Figure 6: SFER for different MCSs.

power of 15 dBm.[1] When the station moves at an average speed of 1 m/s, the optimal length of the A-MPDU maximizing the throughput is 10 subframes (of 1,538 bytes), or equivalently the optimal A-MPDU transmission time equals about 2 ms. In addition, the optimal aggregation time for an average speed of 0.5 m/s is about 2.9 ms where an A-MPDU is composed of 15 subframes. Interestingly, the optimal aggregation time for the average speed of 1 m/s, i.e., 2 ms, is smaller than the coherence time, 3 ms, calculated in Section 3.1.

In order to delve into the aforementioned issues, we investigate the impact of A-MPDU length through additional experiments. For the remaining experiments, we use only AR9380 NIC since AR9380 has shown more stable MIMO performance and also is easier to play with.

3.3 Impact of A-MPDU Length

Table 1 summarizes the average throughput and SFER for varying aggregation time bound which determines the aggregation length of A-MPDU. The aggregation time of 0 μs represents the transmission of a single MPDU without aggregation. As the length of A-MPDU increases, the throughput of the static scenario (0 m/s) increases due to the overhead reduction, but for an average speed of 1 m/s, the maximum throughput is achieved at 2,048 μs aggregation time bound as calculated in Section 3.2. When the aggregation time bound is larger than 2,048 μs, the increased SFER induced by the mobility overwhelms the gain from the overhead reduction. Accordingly, the throughput decreases as the aggregation time bound increases.

Table 2: MCS information

	MCS 0	MCS 2	MCS 4	MCS 7
Modulation	BPSK	QPSK	16-QAM	64-QAM
Code rate	1/2	3/4	3/4	5/6
Data rate (Mbit/s)	6.5	19.5	39	65

3.4 Impact of MCSs

MCS is a critical factor determining the receiving performance as described in Section 2.2.2. Although higher order MCS achieves higher data rate, it is vulnerable not only to

[1]We translate BER into SFER according to the subframe location in Figure 5(b), and numerically calculate the achievable throughput employing A-MPDU for a given number of subframes.

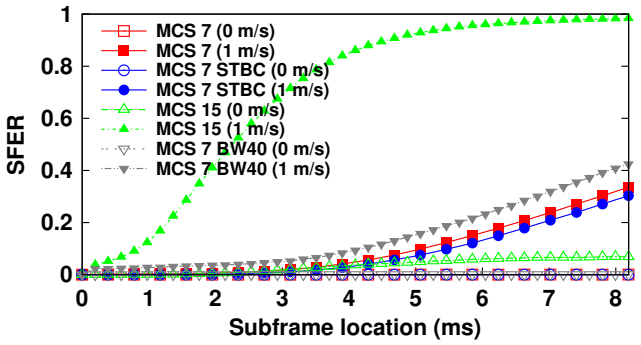


Figure 7: SFER with various 802.11n features.

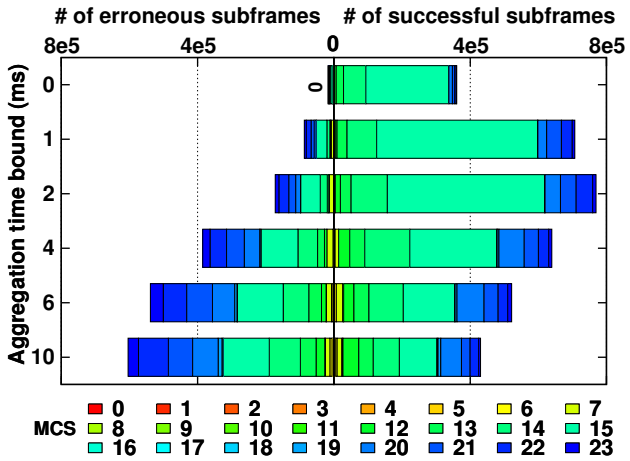


Figure 8: Rate distribution on *Minstrel*.

the channel degradation but also to mobility, since constellation points are closer from each other and coding gain is generally smaller. To verify this, we conduct experiments by changing MCSs for a given mobility. Table 2 summarizes the detailed information of MCSs used in this measurement.

Figure 6 shows the SFER depending on the subframe location. Unlike the evaluation of previous sections, we plot SFER instead of BER for better intuition. When a station holds its position at P1, the SFER remains almost zero in all subframe locations regardless of the employed MCS, because the channel quality between AP and the station is considerably good. However, when the station moves at an average speed of 1 m/s, MCS 4 and MCS 7 employing both amplitude and phase modulation (i.e., 16-QAM and 64-QAM, respectively) show higher SFER in the latter part of A-MPDU, while MCS 0 and MCS 2 which use only phase modulation (i.e., BPSK and QPSK, respectively) achieve stable SFER across the entire subframe locations. That is, MCSs which use amplitude modulation are highly susceptible to mobility. An A-MPDU using high order MCS not only uses longer A-MPDU length for a given A-MPDU duration, but also suffers from higher SFER at the latter part of A-MPDU, especially, when the channel varies within the transmission time of A-MPDU. Therefore, the length of A-MPDU with high order MCS should be carefully determined.

3.5 IEEE 802.11n Features

We investigate the impact of IEEE 802.11n features, such as STBC, SM, and channel bonding, in order to verify whether these features alleviate or aggravate the observed problem. We narrow the moving range between P1 and P2 so that the transmitter can utilize double streams (i.e., MCS 15). Figure 7 shows SFER with various 802.11n features. We basically use the results of MCS 7 as a reference for the performance comparison.

First, STBC is well known for offering diversity gain, but we observe that the SFER is only slightly decreased by STBC. That is, STBC cannot suppress the increase of SFER in the latter part of A-MPDU.

With SM, which is denoted as MCS 15 (using two spatial streams with 64-QAM and 5/6-rate code), the SFER is further deteriorated. Even if the station does not move (i.e., MCS 15 with 0 m/s curve), the SFER grows as the subframe location number increases, since MIMO requires a more accurate channel compensation to eliminate the spatial interference. When the station moves, only a few subframes in the beginning part of A-MPDU can be successfully transmitted.

The result of 40 MHz bandwidth shows slightly higher SFER than that of 20 MHz bandwidth. That is because not only the power per bandwidth is decreased but also the channel compensation is more difficult than 20 MHz bandwidth due to the increased number of subcarriers.

In all the above cases, the 802.11n features do not solve the observed problem; SM and 40 MHz channel bonding are more severely affected by mobility. Moreover, STBC does not alleviate the performance degradation induced by mobility either.

Table 3: Throughput and SFER on *Minstrel*

Time bound (μs)	0	1,024	2,048	4,096	6,144	10,240
Throughput (Mbit/s)	37.54	73.12	79.83	66.38	53.64	44.24
SFER (%)	5.13	11.92	19.42	37.77	50.87	58.33

3.6 Rate Adaptation: *Minstrel*

Rate adaptation algorithm (RA) selects the most suitable MCS (or PHY rate) for a given channel condition. Every 802.11 device implements a RA to maximize the performance (e.g., to maximize the throughput) while these algorithms are implementation dependent. We use a measurement to investigate the impact of mobility when a RA runs. *Minstrel* is used in this experiment, since it is well known for achieving good performance in practical environments [19] and Linux wireless tool adopts it as the default embedded RA [6]. *Minstrel* is a window-based RA, which collects the wireless channel statistics by transmitting probing frames, i.e., data frames transmitted with a randomly selected PHY rate. The ratio of probing frames is generally 10% of the total traffic. Based on that, *Minstrel* calculates the best throughput rate within a time window, and adopts it as a basic rate for the next window.

Figure 8 illustrates the MCS distribution measured for a mobility scenario with varying aggregation time bound. Each stacked bar represents the number of subframes transmitted with a specific MCS, where left-side and right-side correspond to the numbers of erroneous and successful subframes, respectively. Probing frames are not counted in the statistics since they are not aggregated into A-MPDU. When

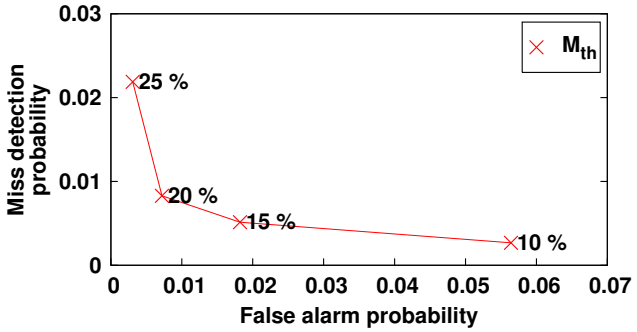


Figure 9: Accuracy of MD.

A-MPDU is not used, there are few frame errors. However, as the aggregation time bound increases, the SFER also increases, especially, it rises steeply between 2 ms and 4 ms. Consequently, the maximum throughput is achieved with 2 ms aggregation time bound, using mostly MCS 15.

We observe that *Minstrel* tries to use higher order MCSs over MCS 15 more often, when the aggregation time bound becomes larger than 2 ms. The reason of this bad MCS selection is caused by the fact that SFER using the current MCS increases, because subframes in the latter part of A-MPDU fail due to mobility. However, frame error rate (FER) of probing frames might be lower than that of the currently employed MCS, since probing frames are sent to the air without aggregation, even if the MCS used by probing frames is higher/lower and not suitable. Then, *Minstrel* decides to increase/decrease the PHY rate since the estimated throughput with the MCSs used for probing frames could be larger than that of the current one. As a result, the PHY rate is updated to one of the probed rates, which induces unnecessarily frequent PHY rate variation. Consequently, the throughput is further deteriorated when the RA runs in mobile environments.

More detailed information is summarized in Table 3. As expected, the maximum throughput is achieved with 2 ms aggregation time bound, which achieves the optimal trade-off between the overhead reduction from aggregation and the SFER increment by mobility.

4. PROPOSED ALGORITHM DESIGN

Section 3 has demonstrated the inherent limitation of the 802.11n frame aggregation schemes. Motivated by this observation, we develop MoFA, a novel frame aggregation algorithm, composed of *mobility detection* (MD) and *A-MPDU length adaptation* in order to overcome the limitation. In addition, MoFA adopts an adaptive use of RTS/CTS to suppress the hidden collision which leads to an ill behavior of MoFA otherwise.

4.1 Mobility Detection

Mobility leads to SFER increase at subframes in the latter part of A-MPDU, while SFER caused by the poor channel (i.e., the channel in low SNR regime) is uniformly distributed irrespective of subframe location. Taking into account the different characteristics, we employ MD scheme into MoFA to determine the degree of the mobility. Let N be the number of aggregated subframes in an A-MPDU. $\overrightarrow{S} = \{s_1, s_2, \cdots, s_N\}$, updated after receiving a BlockAck, represents the transmission results of subframes, where s_i is equal to 1 or 0 depending on whether the corresponding ith subframe succeeds or fails, respectively. Then, the average SFERs of front and latter halves of an A-MPDU are respectively represented by:

$$\text{SFER}_f = \frac{\sum_{i=1}^{N_f} s_i}{N_f} \quad \text{and} \quad \text{SFER}_l = \frac{\sum_{i=N_f+1}^{N} s_i}{N - N_f}, \tag{3}$$

where $N_f = \lfloor N/2 \rfloor$.

We quantify the degree of the mobility, M, by using SFER_f and SFER_l, defined as follows:

$$M = \text{SFER}_l - \text{SFER}_f. \tag{4}$$

Therefore, if M is larger than the mobility detection threshold, M_{th}, MD scheme determines that the station is under a mobile environment.

To verify the accuracy of MD, we conduct experiments by assessing miss detection and false alarm probabilities. We use the same experiment settings as that described in Section 3.2, except that MD scheme is implemented in the AP. Figure 9 shows that as M_{th} increases, the false alarm probability decreases, while the miss detection probability increases. It is because high M_{th} effectively differentiates the loss caused by a poor channel from the loss caused by mobility, but it cannot detect mobility which causes relatively low M. Finding an appropriate M_{th} is important in order to achieve an accurate MD operation according to the degree of the mobility. Based on the measurement results, we observe that $M_{th} = 20\%$ results in suitable miss detection and false alarm probabilities. Therefore, we fix M_{th} to 20% throughout the rest of the paper.

4.2 A-MPDU Length Adaptation

We now propose an A-MPDU length adaptation algorithm to find an appropriate length with respect to the degree of the mobility.

4.2.1 A-MPDU length decrease

The length of A-MPDU should be decreased to maximize the throughput in a mobile environment. Given sufficient statistics, it is possible to find out the optimal aggregation time bound of A-MPDU. Assuming that the payload size is fixed, let N_t be the maximum number of possible aggregated subframes for a given aggregation time bound T_o. That is,

$$\begin{aligned} N_t &= \underset{n}{\text{argmax}}(n \cdot L/R + T_{oh}), \\ &\text{s.t. } n \cdot L/R + T_{oh} \le T_o, \end{aligned} \tag{5}$$

where L is the subframe length including the overhead of A-MPDU delimiter and its padding bits. R represents the current PHY rate, and T_{oh} denotes the time overhead including DIFS, backoff, PLCP preamble, PLCP header, SIFS, and transmission time of a BlockAck.

$\overrightarrow{P} = \{p_1, p_2, \cdots, p_{N_t}\}$ represents the SFER statistics of each subframe, where p_i stands for the SFER of the ith subframe, which is calculated using an exponential weighted moving average (EWMA). $\overrightarrow{P}$ is updated at every BlockAck reception, and accordingly, each element is denoted by:

$$p_i := \begin{cases} (1-\beta)p_i + \beta, & \text{if fails,} \\ (1-\beta)p_i, & \text{if succeeds,} \end{cases} \tag{6}$$

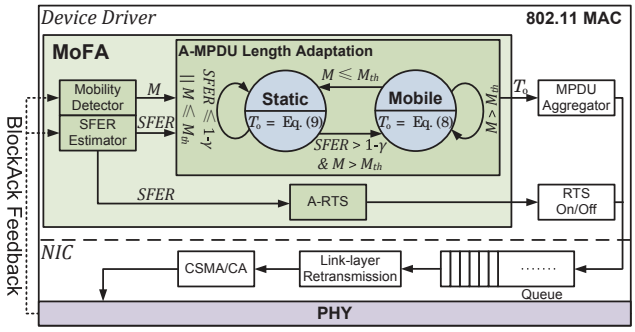


Figure 10: Operation of MoFA.

where β is a weighting factor, which implies that most recent sample carries β weight in the estimation. Throughout the rest of the paper, we set β in Equation (6) to 1/3, which implies the most recent transmission result carries 1/3 weight in the estimation. Meanwhile, using $\overrightarrow{P}$, the optimal length of A-MPDU, n_o, can be calculated as follows.

$$n_o = \underset{n \le N_t}{\operatorname{argmax}} \frac{\sum_{i=1}^{n} L(1-p_i)}{n \cdot L/R + T_{oh}}. \quad (7)$$

Then, N_t is updated to n_o, and finally, the aggregation time bound T_o is determined as follows:

$$T_o := n_o \cdot L/R + T_{oh}. \quad (8)$$

Note that the newly updated T_o is less than or equal to the previous value because $n_o \le N_t$. By adjusting the aggregation time bound of A-MPDU based on the statistics, MoFA limits the number of aggregated subframes in order to achieve the optimal throughput.

4.2.2 A-MPDU length increase

Generally, longer A-MPDU improves the channel efficiency in static environments as shown in Section 3. Therefore, when $M \le M_{th}$, the current channel is very likely ready to support a longer A-MPDU, and hence, the number of subframes should be increased.

In order to increase the length of A-MPDU, MoFA increases the aggregation time bound, T_o:

$$T_o := \min\left(T_o + n_p \cdot L/R, T_{max}\right), \quad (9)$$

where T_{max} is the maximum possible PPDU transmission time, 10 ms, and n_p is the number of additionally allowed subframes, which are referred to as probing subframes. The amount of increment varies according to the PHY data rate, since it is inversely proportional to the current PHY rate, R. To achieve rapid length adaptation of A-MPDU keeping up with the channel dynamics, n_p is determined by an exponential increment, i.e., $n_p = \varepsilon^{n_c}$, where ε is an exponential weighting factor, and n_c denotes the number of consecutive transmissions of A-MPDUs, satisfying that $M \le M_{th}$. As ε becomes larger, A-MPDU length increase algorithm can operate faster by adapting to the wireless channel variation more rapidly. However, large ε can incur large unnecessary probing overhead once the A-MPDU length is properly determined. Here, we set ε to the minimum value, 2, conservatively in order to eliminate such overhead. For example, the numbers of probing frames are 2, 4, and 8 for 3 consecutive increases.

4.3 Adaptive Use of RTS/CTS

The performance of the 802.11 is severely deteriorated when there exist hidden stations. Since some error patterns induced by the hidden collision might not be clearly differentiated from the errors from mobility, i.e., both can induce relatively more errors in the latter part of A-MPDUs, the above schemes might not work properly in the presence of hidden stations. Therefore, to ameliorate such a hidden station problem, we employ an adaptive use of Request-To-Send (RTS) and Clear-To-Send (CTS) exchange. However, the use of RTS/CTS procedure without a prudent decision can degrade the performance of WLAN due to its overhead. Adaptive uses of RTS/CTS have been proposed recently [18, 13].

In this paper, we improve the concept of *adaptive RTS* (A-RTS) *filter* proposed in [18], which selectively turns on RTS transmission per data frame by adapting the dynamic collision level incurred by hidden stations. Different from the original A-RTS, which does not consider A-MPDU operation, MoFA decides when and how RTS should be turned on or off, according to the transmission result of an A-MPDU.

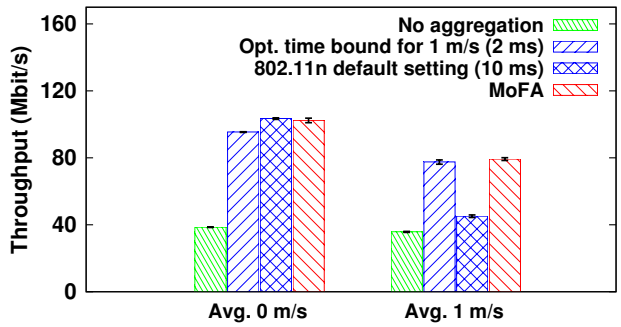
The key state of A-RTS in MoFA is *RTSwnd*, i.e., the window size of RTS, representing the number of consecutive A-MPDU frames that will use an RTS/CTS exchange before the transmission. *RTScnt*, which is set to *RTSwnd* whenever *RTSwnd* value is updated, keeps track of the remaining number of frames to be sent after an RTS/CTS exchange. That is, an RTS/CTS exchange is enabled before a data transmission as long as $RTScnt > 0$. *RTSwnd* is set to 0 initially, and adaptively increases whenever a collision is suspected. *RTSwnd* increases by one when the instantaneous SFER (*SFER*) of the last A-MPDU transmitted without RTS is larger than $1-\gamma$, where γ denotes SFER threshold. *RTSwnd* also employs multiplicative decrease to minimize the RTS/CTS exchange overhead; if $SFER > 1-\gamma$ with RTS or $SFER \le 1-\gamma$ without RTS, *RTSwnd* becomes a half, since RTS/CTS exchange seems not useful.[2]

4.4 Operation of MoFA

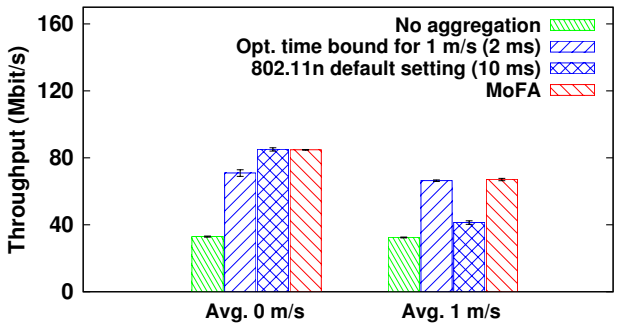
In this section, we explain how the aforementioned three components of MoFA, i.e., MD, A-MPDU length adaptation, and A-RTS, interwork together. The detailed procedure of MoFA is best explained with a blueprint in Figure 10. The state transitions of A-MPDU length adaptation depend on two MoFA parameters, namely, (1) the degree of the mobility (M) and (2) the instantaneous SFER (*SFER*). Upon the reception of a BlockAck, MoFA estimates both M and *SFER* by using MD and SFER estimator, respectively.

A-MPDU length adaptation differentiates two states, namely, **static** and **mobile**. If the frame errors appear insignificant (i.e., $SFER \le 1-\gamma$), or if the frame errors seem caused by poor channel (i.e., $M \le M_{th}$), MoFA stays in the **static** state. In other words, MoFA decides not to decrease the

[2]After receiving a BlockAck, *SFER*, i.e., the number of transmission-failed subframes divided by the total number of subframes within the most recently transmitted A-MPDU, can be obtained. If BlockAck is not received after an A-MPDU transmission, $SFER := 1$.



(a) Transmit power: 15 dBm.



(b) Transmit power: 7 dBm.

Figure 11: Throughput in one-to-one scenario.

aggregation time bound T_o, but tries to increase the bound according to Equation (9).

However, if $SFER > 1 - \gamma$ and MD detects that the loss caused by mobility (i.e., $M > M_{th}$), the state switches to the **mobile** state. In this state, MoFA decreases T_o according to Equation (8) until the reception of subframes in the latter part of A-MPDU is not affected by mobility (i.e., $M \leq M_{th}$). While obtaining the appropriate A-MPDU length, A-RTS operates independently and simultaneously to adapt the dynamic collision level as described in Section 4.3.

Considering both the overhead of RTS/CTS exchange and the trade-off between the loss induced by mobility and the benefit of amortizing the overhead using A-MPDU, selecting the appropriate γ is an open-problem, which remains as the future work. In the rest of the paper, we set SFER threshold, γ, to 0.9 as a rule of thumb, and hence, 10% subframe error in A-MPDU triggers A-MPDU length adaptation in mobile environments [20].

Note that MoFA works independently from RAs. Instead, it helps RAs not to be misled by the errors caused by mobility. Specifically, MoFA dynamically adjusts the length of A-MPDU according to the degree of the mobility and suppresses the mobility-induced excessive subframe errors, and hence, it can prevent malfunction of rate control algorithms.

5. EXPERIMENTAL EVALUATION

In this section, we comparatively evaluate the performance of MoFA measured under a wide range of scenarios. We have implemented MoFA on an off-the-shelf 802.11 NIC with AR9380 by modifying the *ath9k* device driver. Section 2.3 describes the details of the measurement setting for this evaluation. For each set of experiment scenarios, we average the results of 5 runs, where each lasts for 60 seconds.

5.1 One-to-One Scenario

For comparison, we apply the optimal fixed time bounds for both static and mobile stations, which we have verified in Section 3.3. The optimal fixed time bound of the static station is the maximum time bound, 10 ms, which is 802.11n default setting. On the other hand,the optimal fixed time bound is 2 ms for an average speed of 1 m/s.

5.1.1 Static vs. mobile environments

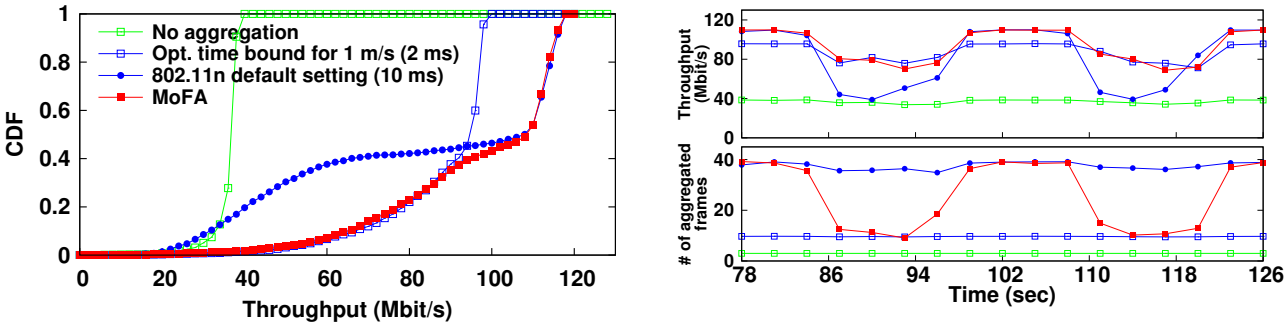
We evaluate the performance of MoFA in both static and mobile environments, where a single station stays or moves consistently during the experiment. Figure 11 shows the throughput for the given degree of the mobility, with error bars representing the standard deviation. We use 15 dBm and 7 dBm transmit powers in order to investigate the impact of the channel quality. We observe that if the aggregation is disabled, the throughput does not vary with the average speed since it is scarcely affected by the degree of the mobility. When the station does not move (i.e., average 0 m/s), 802.11n default setting shows the best throughput in both transmit powers as expected in Section 3. Using 15 dBm transmit power, the required transmission time of the longest A-MPDU is much lower than 10 ms.[3] Therefore, the achieved throughput of optimal fixed time bound for 1 m/s is 7.8% lower compared to the best throughput. However, when 7 dBm transmit power is used, it shows 17.7% lower throughput than that of 802.11n default setting as shown in Figure 11(b) because of the marginal overhead growth. MoFA tends to use the longest A-MPDU in stationary environment, and hence, achieves the maximum throughput for both transmit power cases.

On the other hand, when the station moves with an average speed of 1 m/s, the throughput of 802.11n default setting is deteriorated due to the limitation of the channel compensation. Interestingly, MoFA achieves the highest throughput, surpassing the optimal fixed time bound for 1 m/s, by adjusting the length of A-MPDU according to the degree of the mobility. Specifically, the degree of the mobility changes instantaneously, even though its average value does not vary. As a result, MoFA achieves 2.2% and 1.1% higher throughput than that of the optimal fixed time bound (= 2 ms) for 15 dBm and 7 dBm transmit power, respectively. Since 802.11n default setting sets longer length of A-MPDU as the optimal fixed time bound, it results in the increase of SFER at the latter part of A-MPDUs, which causes the throughput degradation. The gains of MoFA from 802.11n default setting are 75.6% and 62.4% for 15 dBm and 7 dBm transmit power, respectively. The reasons why the gain drops as the transmit power decreases, comes from the use of reliable modulation and code rate against the mobility.

5.1.2 Time-varying mobile environment

Here, we investigate the adaptability of MoFA, i.e., A-MPDU length adaptation capability as the degree of the

[3]Using high order MCS, A-MPDU length is limited not by the time bound but by the maximum A-MPDU length, 65,535 bytes.



(a) CDF of instantaneous throughput.

(b) Performance over time.

Figure 12: Performance of MoFA in time-varying mobile environment.

mobility changes over time. Figure 12(a) shows the empirical CDF of the instantaneous throughput for 200 ms. A station stays and moves half-and-half with a regular pattern. Therefore, 50% of the samples are exposed in mobile environments and the others are collected in static environments.

When frame aggregation is disabled, over 90% of the samples are in the range of 35 to 38 Mbit/s stably, since mobility does not affect the reception of a single non-aggregated frame. On the other hand, enabling A-MPDU, each curve is divided into two regions. Especially, the CDF from 0 to 0.5 corresponds to mobile environments, because the performance of A-MPDU is degraded in mobile environments as shown in Section 5.1.1. In this region, 802.11n default setting shows the worst performance where almost 40% of samples are below 60 Mbit/s. Meanwhile, the optimal fixed time bound for average speed of 1 m/s performs well by limiting the length of A-MPDU up to 2 ms. In the case of MoFA, its curve reaches up to the most outer curve which is obtained by the optimal fixed time bound for 1 m/s . Turning to the result of the static environment, the CDF from 0.5 to 1, 802.11n default setting shows the highest throughput apparently, and MoFA shows almost the same result, trying to use the longest A-MPDU.

Figure 12(b) demonstrates the trace of instantaneous throughput and number of aggregated frames over time. MoFA tries to use the maximum A-MPDU length in static environments while it strives to use short A-MPDUs, close to the optimal fixed time bound in mobile environments. Therefore, the instantaneous throughput of MoFA follows the upper-most curves. Interestingly, the number of aggregated subframes with 802.11n default setting slightly fluctuates in mobile environments. In fact, a BlockAck can acknowledge up to 64 consecutive MPDUs with a fixed-size bitmap. Accordingly, the difference between the sequence numbers of the first and the last MPDUs aggregated in a single A-MPDU frame should be less than 64. If the (re)transmissions of the first subframe continues to fail, then the maximum number of aggregated frames could decrease due to the constraint.

In summary, none of fixed time bound settings can achieve the best throughput all the time in time-varying mobile environment. However, MoFA, which adaptively changes the A-MPDU length according to the degree of the mobility, can approach the optimal throughput over the entire experiments. As an evidence, we have shown that the instantaneous throughput of MoFA follows the outer-most curves irrespective of the mobility level.

5.1.3 *Hidden terminal environment*

We now investigate how A-RTS in MoFA works in the presence of hidden terminals. Figure 13 presents the throughput for both static (with three kinds of hidden source rates) and mobile scenarios. A hidden AP is located at P7 and it sends downlink UDP traffic to a station located at P6. In the case of static scenario, a target station settled at P4 can carrier-sense both APs, but APs cannot detect the signal of each other. We compare MoFA against 802.11n default setting, 10 ms, since it is the optimal for static environments. For a fair comparison, we also use RTS/CTS exchange for the maximum time bound.[4]

In the absence of hidden traffic (i.e., 0 Mbit/s), the throughput of the optimal fixed time bound with RTS is slightly lower than the maximum throughput due to RTS/CTS overhead, but it shows the best performance as hidden source rate increases. Thanks to the A-RTS, MoFA achieves the high throughput close to the maximum throughput regardless of hidden source rate. MoFA enables RTS/CTS exchange before most A-MPDU transmissions when there exists hidden terminals, and vice versa.

We next investigate the performance of MoFA in the presence of both the hidden interference and the mobility. That is, a target station moves between P3 and P4 with an average speed of 1 m/s under the hidden interference from P7. We compare MoFA against the optimal fixed time bound for average speed of 1 m/s (2 ms), which shows the best performance for the mobile environments. Apparently, the optimal fixed time bound with RTS performs the best, because it aggregates average 10 subframes into a single A-MPDU scarcely affected by the mobility, and enables RTS/CTS exchange all the time to handle the hidden terminal problem effectively. The throughput of MoFA slightly decreases by 5.85% compared with the maximum throughput because of overlapping of both the miss detection/false alarm of MD and A-RTS.

5.2 Multiple Node Scenario

We finally evaluate the performance of MoFA in a multi-node environment. Five nodes and one AP are deployed with the floor plan as illustrated in Figure 4. The AP sends fully saturated downlink traffic to the each station. Three nodes (i.e., STA1, STA2, and STA3) move between P1 and P2,

[4]RTS/CTS exchange is always conducted before transmitting an A-MPDU. We refer to this as the 'optimal fixed time bound with RTS.'

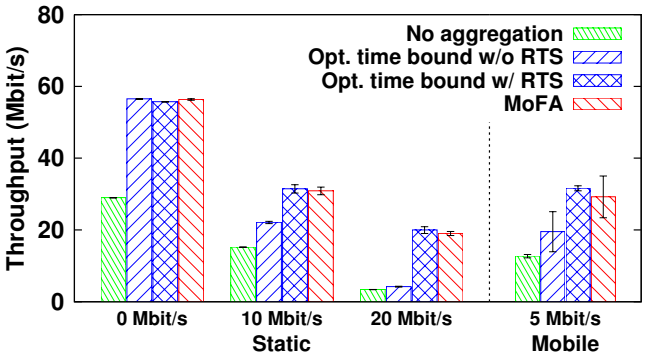


Figure 13: Throughput with hidden terminals.

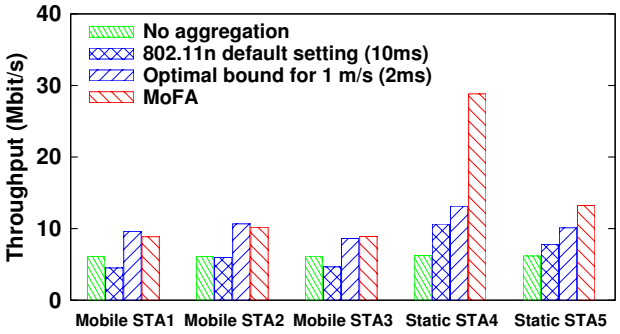


Figure 14: Throughput with multiple nodes.

P8 and P9, P3 and P4, respectively. STA4 and STA5 hold their positions at P5 and P10. We compare MoFA against no aggregation scheme, 802.11n default setting (= 10 ms), and the optimal fixed time bound for mobile scenarios.

Figure 14 shows the downlink throughput achieved by each station, with error bars representing the standard deviation. When frames are transmitted without aggregation, each station's achieved throughput is almost uniform because IEEE 802.11 MAC basically provides an equal opportunity for the channel access to all the contending stations in the long term. On the other hand, enabling A-MPDU provides different throughput for each station depending on the channel dynamics, since stations can transmit multiple MPDUs back-to-back without additional channel access procedures. Interestingly, the stationary STA4 achieves the biggest gain by using MoFA, which is counter intuitive. Main reason of this observation comes from the fact that stations share the wireless medium together. MoFA tends to use relatively short length of A-MPDUs for mobile stations by detecting mobility, which leads to low SFER, and hence, it prevents the waste of resource. All stations then share the additionally acquired resource, and especially, STA4 which is stationary and close to the AP, gets the biggest benefit thanks to the use of longer A-MPDUs. Overall, MoFA achieves 127%, 109%, and 35% higher network throughput than no aggregation, 802.11n default setting, and the optimal fixed length for mobile scenarios, respectively.

As we have observed so far, the overall performance of MoFA highly depends on the network topology and the degree of mobility of each station. We expect that the significant growth of Wi-Fi devices will lead to much higher degrees of mobility and hidden interferences, and hence, MoFA will highly benefit future large-scale Wi-Fi networks.

6. RELATED WORK

Numerous frame aggregation algorithms have been proposed to enhance the MAC efficiency, primarily based on numerical analysis and simulations (with unrealistic assumptions). Many papers have proposed the optimal frame length under error-prone channels. An analytic model for estimating the performance of A-MPDU and A-MSDU is presented in [9] to show that A-MSDU considerably degrades the performance as the aggregation length increases, while longer A-MPDU shows better performance over an erroneous channel. The optimal frame length adaptation algorithm for A-MSDU is proposed in [15], which provides a packet loss model, which includes collisions and channel fading.

The algorithms in [8, 11] use A-MPDU and A-MSDU simultaneously, and determine the A-MPDU subframe length in the same manner as the A-MSDU length optimization. In [11], He *et al.* find the optimal A-MPDU subframe length after rate selection. The proposed algorithm in this work utilizes the pre-designed mapping table which gives the optimal MPDU size according to BER input. Thanks to the mapping table, when the BER of the current wireless channel is given, the length of A-MPDU subframe is easily determined. The work in [8] also uses the mapping table, which provides the optimal MCS and MPDU size simultaneously, based on the SNR of the received signal. However, all these studies are based on the unrealistic assumption that all A-MPDU subframes experience the uniform SFER. Consequently, they are not concerned with finding the optimal A-MPDU length.

Robust channel estimation techniques for frames requiring long transmission time are proposed in [10, 14]. The former injects mid-amble which plays the same role as PLCP preamble, i.e., allowing the receiver to learn the channel, in the middle of A-MPDU reception while the latter develops scattered pilot technique which reorganizes the pilot pattern periodically. However, these approaches are not standard-compliant, thus making them costly and impractical for large-scale adoption by commercial products. In contrast, our approach fully complies with the current 802.11n standard which serves as a main advantage over others for wide deployment in the real world.

7. CONCLUDING REMARKS

In IEEE 802.11n WLAN, we reveal that using long A-MPDU frames cuts the throughput by up to two thirds regardless of the channel quality at the receiver in time-varying channel environments. Motivated by that observation, we develop MoFA, a novel standard-compliant algorithm, which dynamically adapts the length of the A-MPDU in run-time, by observing the mobility-caused channel dynamics in order to maximize the network performance. Our prototype implementation in commercial IEEE 802.11n Wi-Fi devices demonstrates that MoFA achieves throughput 1.8x higher than the default setting of 802.11n in mobile environments. To our best knowledge, this is the first effort to optimize the A-MPDU length for commercial IEEE 802.11n devices. Joint optimization of the length of A-MPDU and rate adaptation will be included in our future work.

Acknowledgment

This research was supported by LG U+ project, "Research on WLAN management and operation technologies for real-time service in condensed WLAN environments," and the Brain Korea 21 Plus Project in 2014.

8. REFERENCES

[1] Intel WiFi Link 5300. http://www.intel.com/products/wireless/adapters/5000/.

[2] Ath9k: Atheros Linux Wireless Driver. http://wireless.kernel.org/en/users/Drivers/ath9k/.

[3] Iperf: TCP/UDP Bandwidth Measurement Tool. http://dast.nlanr.net/Projects/Iperf/.

[4] Linux 802.11n CSI Tool. http://dhalperi.github.io/linux-80211n-csitool/.

[5] HostAP: IEEE 802.11 AP, IEEE 802.1X/WPA/WPA2/EAP/RADIUS Authenticator. http://hostap.epitest.fi/hostapd/.

[6] Linux Wireless Tools. http://linuxwireless.org/.

[7] Bhartia, A., Chen, Y. C., and Qiu, L. Harnessing Frequency Diversity in Wi-Fi Networks. In *Proc. ACM MobiCom'11* (Sept. 2011).

[8] Feng, K., Lin, P., and Liu, W. Frame-Aggregated Link Adaptation Protocol for Next Generation Wireless Local Area Networks. *EURASIP Journal on Wireless Communications and Networking* (2010).

[9] Ginzburg, B., and Kesselman, A. Performance Analysis of A-MPDU and A-MSDU Aggregation in IEEE 802.11n. In *Proc. IEEE Sarnoff Symposium* (May 2007).

[10] Grunheid, R., et al. Robust Channel Estimation in Wireless LANs for Mobile Environments. In *Proc. IEEE VTC'02 Fall* (Sept. 2002).

[11] He, X., Li, F. Y., and Lin, J. Link Adaptation with Combined Optimal Frame Size and Rate Selection in Error-Prone 802.11n Networks. In *Proc. IEEE ISWCS'08* (Oct. 2008).

[12] IEEE Std. *IEEE 802.11, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, Mar. 2012.

[13] Kim, J., Kim, S., Choi, S., and Qiao, D. CARA: Collision-Aware Rate Adaptation for IEEE 802.11 WLANs. In *Proc. IEEE INFOCOM'06* (Apr. 2006).

[14] Kim, S. I., Oh, H. S., and Choi, H. K. Mid-amble Aided OFDM Performance Analysis in High Mobility Vehicular Channel. In *Proc. IEEE IV'08* (June 2008).

[15] Lin, Y., and Wong, V. W. Frame Aggregation and Optimal Frame Size Adaptation for IEEE 802.11n WLANs. In *Proc. IEEE GLOBECOM'06* (Nov. 2006).

[16] Perahia, E. and Stacey, R. Next Generation Wireless LANs: Throughput, Robustness, and Reliability in 802.11n. Cambridge University Press, 2008.

[17] Steele, R. *Mobile Radio Communications*, first ed. IEEE Press, 1995.

[18] Wong, S. H. Y., et al. Robust Rate Adaptation for 802.11 Wireless Networks. In *Proc. ACM MobiCom'06* (Sept. 2006).

[19] Yin, W., Bialkowski, K., Indulska, J., and Hu, P. Evaluation of MadWifi MAC Layer Rate Control Mechanisms. In *Proc. IEEE IWQoS'10* (June 2010).

[20] Zhang, J., Tan, K., Zhao, J., Wu, H., and Zhang, Y. A Parctical SNR-Guided Rate Adaptation. In *Proc. IEEE INFOCOM'08* (June 2008).