Computer Communications 34 (2011) 1316–1327

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

REACT: Rate Adaptation using Coherence Time in 802.11 WLANs

Hakyung Jung, Ted "Taekyoung" Kwon *, Kideok Cho, Yanghee Choi

School of Computer Science and Engineering, Seoul National University, Seoul 151-742, Republic of Korea

ARTICLE INFO

Article history:
Received 29 December 2009
Received in revised form 17 December 2010
Accepted 24 January 2011
Available online 3 February 2011

Keywords:
Wireless LAN
Rate adaptation
Multi-rate
802.11
Coherence time

ABSTRACT

The channel coherence time in indoor WLANs normally exceeds multiple frame transmission times. In light of this, we propose a new rate adaptation scheme, termed as the *Rate Adaptation using Coherence Time* (*REACT*), that has the following key features. First, without exchanging RTS and CTS frames, the receiver in REACT informs the transmitter of the improved channel condition via altering the ACK transmission rate, so that the transmitter increases the data rate for subsequent data frames. This enables the transmitter to adapt to the time-varying channel conditions while inducing the marginal overhead. Second, the transmitter in REACT can identify the reasons of frame losses by exploiting the feedback from the receiver and the estimated coherence time. Frame losses are assumed to be caused only by collisions for the duration of the coherence time after receiving an ACK frame with the altered bit rate. The coherence time is also used to enhance the adaptive RTS probing, so that the REACT can prevent the transmitter from decreasing its bit rate when collisions occur. Extensive simulations reveal that REACT consistently performs better than the other rate adaptation schemes (ARF, CARA, RRAA, and RBAR) in all the testing scenarios.

1. Introduction

The 802.11 standards specify multiple transmission rates at the physical layer by employing different modulation schemes and error correction codes. Vendors and researchers have introduced a number of proposals as the 802.11 standards do not mandate the procedures of selecting the transmission rate [1–19], which are collectively referred to as the *rate adaptation*. In principle, the higher is the received signal-to-noise ratio (SNR), the higher transmission rate can be used for communications. On the other hand, as the received SNR becomes lower, the error-resilient lower transmission rates will be more desirable. In fact, in order to maximize the system throughput, a station must adjust its transmission rate as a reaction to the dynamically changing environments.

Although the optimal transmission rate can be determined when the value of the received SNR is given, making the decision on the optimal rate is still challenging and complicated especially in WLANs due to the lack of channel status feedback. The 802.11 specification neither defines a separate channel for delivering the channel feedback information nor states any feedback mechanisms that need to be sent from the receiver to the transmitter. This makes the rate adaptation of WLANs non-trivial compared with cellular networks that separately employ pilot or control channels. Moreover, if the radio environment rapidly changes due to fading and mobility, the channel quality that is seen by both mobile and non-mobile users substantially varies over time.

Consequently, the rate adaptation algorithms in the 802.11 WLANs either make the transmitter locally estimate the channel status observed at the receiver without the status feedback (called the *open-loop* approach) or use some portion of its bandwidth for delivering the control messages for feedback (called the *closed-loop* approach). The open-loop approach heuristically infers the channel status of a receiver without the feedback information. Therefore, it cannot accurately estimate the channel (especially when frames are lost due to collisions), and cannot quickly react to the channel variation because it requires some transmission history to estimate the channel. The closed-loop approach can quickly react to the channel variation by exploiting the accurate feedback from a receiver. However, it usually suffers from control overhead, i.e., delivering feedback may severely waste precious bandwidth in certain scenarios. Furthermore, the proposed frame formats do not conform to the 802.11 standard, and this hinders its practical deployment.

In this paper, we introduce a novel rate adaptation scheme, called the *Rate Adaptation using Coherence Time* (*REACT*), which has mainly two key features. First, the receiver in REACT informs the transmitter of the improved channel condition via altering the ACK transmission rate. The channel status information obtained via the preceding ACK frame will be valid for the following data frames because the channel coherence time in WLANs typically exceeds multiple frame transmission times. Upon receiving an ACK frame indicating the good channel condition, the

---

* Corresponding author. Tel.: +82 2 880 9105.
E-mail address: tkkwon@snu.ac.kr (T.T. Kwon).

0140-3664/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2011.01.011

transmitter increases the data rate to the next higher rate. Second, REACT identifies the reason of frame losses by exploiting the feedback from the preceding ACK frame and the coherence time. After receiving an ACK frame indicating the improved channel condition, the transmitter can assume that the channel at the receiver will be favorable for the higher bit rate during the interval of the coherence time. Thus, the data frames that are lost during this interval are deemed to be lost due to occurrence of collisions, and not by channel errors.

We perform comprehensive simulations to evaluate the performance of the proposed scheme. We compare five rate adaptation algorithms, i.e., ARF [1], CARA [4], RRAA [7], RBAR [13], and REACT. Our results show that REACT consistently performs better than the other algorithms in all the simulated scenarios with varying the distance, the number of contending stations, frame size, indoor mobility, and Ricean model parameter.

REACT has the following key advantages.

- Compared with the existing closed-loop schemes, REACT incurs the marginal overhead with respect to RTS/CTS overhead for delivering the channel status information.
- REACT is a receiver-based and responsive approach. When the channel quality is improved, the receiver monitors this change while receiving data frames, and the receiver informs the transmitter of this change at the time of ACK frame transmission via altering the ACK transmission rate.
- REACT can accurately identify the reason of frame losses (whether by channel error or by frame collision) by utilizing the feedback time and the duration of channel coherence time.
- A station that employs REACT can communicate with legacy 802.11 stations since it does not modify the frame formats that are specified by the 802.11 standards; however, other closed-loop schemes modify the frame format.

The remainder of this paper is structured as follows. First, the related work and the background are presented in Section 2. The details of the proposed algorithm and its issues are described in Section 3. Section 4 presents the simulation results. We conclude this paper in Section 5.

2. Related work and background

2.1. Rate adaptation in the 802.11 WLANs

There have been many studies on rate adaptation in 802.11 WLANs. The existing rate adaptation schemes can be categorized into *open-loop* and *closed-loop* approaches, and this depends on whether the channel status feedback from the receiver is delivered to the transmitter or not.

Most of the schemes in the open-loop category estimate the channel quality according to the MAC layer statistics (e.g., [1–10]). In the 802.11 standard, after successfully receiving the data frame, the receiver transmits an ACK frame to the transmitter; thus, the transmitter can regard the frequent (or infrequent) ACK reception failures as signals of a degraded (or improved) channel condition. Each scheme has its own heuristic algorithm to exploit the MAC layer statistics for estimating a channel such as maintaining consecutive success/failure counters [1–5], estimating an expected transmission time for each bit rate [6], calculating a frame loss ratio for recent trials [7], and so on. Other schemes in this category decide the transmission rate based on the local physical layer measurement which is done while ACK frames are received [8–12].

The most promising characteristics of the open-loop schemes is that their operations conform to the 802.11 specification. Thus, most algorithms that are implemented by vendors fall into this category. However, open-loop solutions are inherently hard to quickly react to fast channel dynamics since it takes some delay to collect the MAC layer statistics to estimate the change of the channel condition that requires non-negligible delay. In addition, the open-loop solutions do not perform well in certain scenarios. For example, although [1–3] presume that successive transmission failures are due to poor channel conditions, frame losses can also be caused by interference from hidden stations or collisions [4,7,16,17,19,20]. Moreover, although [8–12] assume that the wireless link between the transmitter and receiver is symmetric, it has been known that wireless links are highly asymmetric [21,22], especially in indoor environments [23].

In the closed-loop category, the receiver measures the channel quality at the physical layer and gives back either the desired transmission rate or channel status information to the transmitter via non 802.11-compliant frames [13–16]. For example, when the receiver in [13] receives an RTS frame from the transmitter, it determines the desirable transmission rate for the next data frame. Then this information is conveyed via a modified CTS frame. Due to the accurate channel information provided by the receivers, those schemes usually achieve a considerable performance gain when compared with those of the open-loop algorithms. In addition, the close-loop schemes naturally react fast because they do not require some delay to collect the transmission results.

Closed-loop proposals, however, should lose some portion of opportunities to transmit more data because exchanging control frames to deliver the feedback for each data frame can cause huge overhead. Moreover, the overhead becomes more severe in certain conditions where the transmission times of the control frames are comparable with those of data frames [24]. Especially, they require modifying the standard frame formats, which lacks the compatibility with legacy 802.11 stations.

In our previous work [18], we explored the possibility of informing the transmitter of the improved channel condition via altering the ACK transmission rate. Whereas it only focuses on the issue of how to increase the rate, REACT completes a rate adaptation framework by dealing with both when to increase and when to decrease the transmission rate. The notion of using coherence time to evaluate rate adaptation algorithms was presented in [25]. Whereas it only characterizes the coherence time sensitivity of existing schemes, REACT proposes a new rate adaptation scheme leveraging the coherence time of long duration in 802.11 WLANs.

2.2. Coherence time in 802.11 WLANs

Now, we turn our attention to the channel coherence time in 802.11 WLANs. The objective is to verify the conjecture that the coherence time in 802.11 WLANs exceeds multiple frame transmission times, so that the channel status information that is acquired upon receiving of an ACK frame can help determine the transmission rate for the subsequent data frames. The coherence time of a wireless channel is a time interval over which the change of the channel impulse response is considered correlated from its previous value. Among several formulae of the coherence time, we choose a somewhat conservative one in [26][1]:

$$T_c \approx \sqrt{\frac{9}{16 \cdot \pi \cdot f_m^2}} \approx \frac{0.423}{f_m}\,[\mathrm{s}] \tag{1}$$

where f_m is the maximum Doppler spread.

---

[1] As the chosen definition may be too conservative (i.e., the coherence time may be too short), we will later introduce a flexible version of this definition in Section 3.3, which is adjustable depending on the dynamic channel condition.

On the other hand, from [27], let T_p be the expected amount of time to complete the successful transmission of a data frame with considering all the channel states: successful transmission, idle, and collision

$$T_p = T_s + \sigma \frac{1 - P_{tr}}{P_s P_{tr}} + T_o \left(\frac{1}{P_s} - 1 \right) \tag{2}$$

where T_s and T_o are the average times the channel is sensed busy due to a successful transmission and a collision, respectively, P_s is the probability that a transmission occurring on the channel is successful, P_{tr} is the probability that there is at least one transmission in the considered slot time, and σ is a slot time. (Refer to [27] for details.)

Table 1 summarizes the coherence time and the expected time to finish a successful transmission, which are calculated as we vary the mobility of a terminal (v) and the number of contending stations (n). We consider the transmission rate of 6 and 54 Mbps in the 802.11a OFDM PHY. At the pedestrian walking speed of 1 m/s (3.6 km/h) [28], the coherence time is approximately 25.39 ms, and it is reduced to 12.69 ms, and 8.46 ms for the mobility of 2 m/s (7.2 km/h), and 3 m/s (10.8 km/h). In addition, the expected time to successfully transmit a data frame with a 1500 bytes payload at 6 Mbps is 2.23 ms, 2.56 ms, and 2.78 ms as the number of contending stations becomes 1, 5, and 10, respectively. *From this calculation, we can verify that the coherence time in 802.11 WLANs normally exceeds the expected time of the successful transmission of multiple data frames, and this motivates us to exploit the channel status feedback for the subsequent data frames via the preceding ACK frame.*

3. Proposed rate adaptation scheme

In this section, we introduce a new rate adaptation approach, called *Rate Adaptation using Coherence Time* (*REACT*). We first explain a design rationale for the new rate adaptation scheme, and we then describe the details.

3.1. Design rationale

We explain the central ideas of REACT in two aspects: *when to increase* and *when to decrease* the transmission rate.

- **When to increase:** The existing open-loop schemes that locally estimate the channel quality (e.g., maintaining a consecutive success counter or calculating the loss ratio in the estimation window) cannot quickly react to an improved channel condition since they need to receive a number of frames to verify the channel status. On the other hand, the closed-loop schemes that deliver the feedback of the channel information via separate frames can quickly react to the channel dynamics, but they induce the non-negligible MAC overhead. *Therefore, the proposed rate adaptation scheme should be able to timely exploit the channel status information without inducing the excessive overhead of delivering the feedback.*
- **When to decrease:** Switching the rate to the next lower rate after two consecutive transmission failures (e.g., ARF [1]) can quickly catch up the deteriorating channel condition. However, this decision can be misled if the collision losses cannot be differentiated from the channel-error-induced ones when there are either many contending stations or hidden stations in the network. *Therefore, the proposed rate adaptation scheme should be able to identify the reasons of frame losses whether they are due to channel errors or collisions.*

Table 1
Coherence time and expected time for a successful transmission.

v (m/s)	T_c (µs)	n	T_p (µs)	
			6 Mbps	54 Mbps
1	25388.5	1	2226.1	448.4
2	12694.3	2	2320.0	438.3
3	8462.8	3	2411.7	444.5
4	6347.1	5	2556.7	460.7
5	5077.7	10	2784.1	491.0

3.2. When to increase: opportunistic feedback from the receiver

The 802.11 standard requires that the ACK frames be transmitted at the maximum bit rate that is constrained by two rules: (i) the transmission rate of an ACK frame should be less than or equal to that of the preceding data frame, and (ii) the ACK frame is transmitted at a rate selected from the basic rate set [29].[2] For instance, the basic rate set in 802.11a typically comprises 6, 12 and 24 Mbps, and the legacy stations that are receiving a data frame at a rate of 36 Mbps will respond with an ACK frame at a rate of 24 Mbps. We call the ACK rate that conform to the above two rules the *legacy ACK rate* hereafter.

The rationale behind these rules is that the ACK frames should be transmitted as fast as possible while ensuring the error-free receptions of the ACK frames under the current channel conditions. The receiver, however, can transmit an ACK frame at a rate other than the legacy ACK rate. Therefore, the receiver can use an ACK rate other than the legacy ACK rate, which is henceforth referred to as the *altered ACK rate*, to inform the transmitter of its channel status information for the following data frame transmissions.

Exploiting the above trick, the receiver in REACT transmits an ACK frame at the altered ACK rate if the channel condition measured by the received data frame is estimated to be good enough to increase the data transmission rate. If the channel condition is not good enough to raise the transmission rate, then an ACK frame will be transmitted with the legacy ACK rate. The transmitter increases the data transmission rate to the next higher rate or retains it depending on the bit rate of the received ACK frame.

There are two possible options for the altered ACK rate: either the next higher or next lower rate than the legacy ACK rate. If a lower ACK rate than the legacy ACK rate is used, the ACK transmission time will be a little increased. Since the ACK frame is only 14 octets, increasing the transmission time will have a marginal impact on the system performance. Thus, the receivers in REACT primarily choose the next lower rate than the legacy ACK rate as the altered ACK rate. Note that there is no next lower ACK rate available if the legacy ACK rate is 6 Mbps. For these cases, the *next higher rate* than the legacy ACK rate will be used for the altered ACK rate in REACT. One might be concerned that this may result in unsuccessful ACK frame transmissions. However, the size of an ACK frame is far smaller than that of a data frame; thus, with the same bit error rate condition, the ACK frames are likely to be successfully delivered compared to data frames. In summary, the next lower ACK rate is used when the data rate is faster than or equal to 12 Mbps, and the next higher ACK rate is used when the data rate is 6 or 9 Mbps.

The feedback operation of REACT is illustrated in Fig. 1, and it is compared with the ARF scheme. Assume that station S_1 transmits data frames to station S_2, and that the channel condition is time-varying and 802.11a is used for the physical layer. The backoff intervals are omitted for simplicity. Successful transmissions are

---

[2] The 802.11 basic service set (BSS) maintains a basic rate set, which is a list of rates that must be supported by every station joining the BSS. {6 Mbps, 12 Mbps, 24 Mbps} is the set of 802.11a mandatory data rates, so it will be assumed to be the BSS basic rate set in this paper.

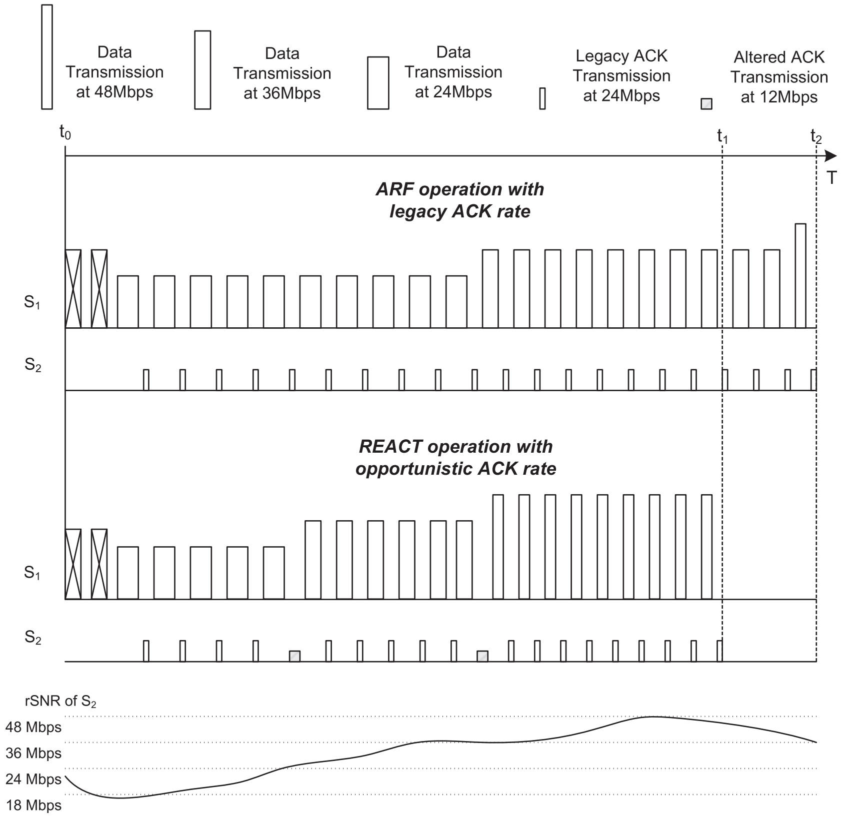


Fig. 1. Illustration of REACT and ARF in the time-varying channel. A station in REACT is able to transmit more frames at higher rates; thus being able to transfer 21 frames earlier than ARF, i.e., $t_1 < t_2$.

depicted by blank rectangles and, transmission failures due to channel errors are depicted by crossed rectangles. At the beginning, both schemes lower the transmission rate to 24 Mbps after experiencing two consecutive transmission failures at the rate of 36 Mbps. After a while (after four data transmissions), the channel quality becomes better so as to allow the transmission at 36 Mbps. However, ARF cannot timely increase the transmission rate because it needs some history of frame statistics (i.e., 10 consecutive successful transmissions) to raise the bit rate. On the other hand, REACT can promptly adapt its rate to the channel by using the feedback from the receiver. In REACT, station S_2 sends an ACK with the altered ACK rate to station S_1 to increase the transmission rate for the subsequent frames. Note that, the feedback mechanism in REACT operates in an *opportunistic* manner. The altered ACK rate is used only once when the channel allows the transmissions of the next higher rate; thus, the feedback overhead is marginal. As a result, a station in REACT is able to transmit more frames at higher rates whereas minimizing the overhead of bandwidth for the feedback.

3.2.1. Issues on the altered ACK rate

The next lower rate. When the next lower rate than the legacy ACK rate is used as the altered ACK rate, the ACK frame transmission time is slightly prolonged. Since the ACK frame is only 14 octets, the increased ACK transmission time is only 4.67 μs (when data rate is faster than or equal to 24 Mbps), or 9.34 μs (when data rate is 12 or 18 Mbps). At the cost of slightly increased ACK transmission time, the throughput gain of REACT is much higher. Assuming that the MAC data payload size is 1500 bytes, the shortened data frame transmission time by increasing data rate ranges from 27.78 μs (when the data rate is increased from 48 Mbps to 54 Mbps) to 666.67 μs (when the data rate is increased from 6 Mbps to 9 Mbps). Since multiple data frames will be delivered at the increased rate, the performance gain of REACT will be even higher. There is a concern that ACK frame transmission might be interfered by other stations accessing the channel as soon as their Network Allocation Vector (NAV) timers expire. In reality, stations can attempt to transmit after the DIFS (34 μs) when frame transmissions has completed. Thus, even hidden stations cannot interfere with the increased ACK transmission.

The next higher rate. When the next higher rate than the legacy ACK rate is used for the altered ACK rate in REACT, one might be concerned that this may result in unsuccessful ACK frame transmissions. However, the size of an ACK frame is far smaller than that of a data frame; thus, with the same channel condition, the ACK frames are likely to be successfully delivered compared to data frames. For example, after receiving a data frame at 6 Mbps, the receiver will respond at the altered ACK rate of 12 Mbps if the channel condition is improved to increase the data rate. Suppose that 12 Mbps is used for both data and ACK frame transmission and MAC data payload length is 1500 bytes. According to [10], the expected frame error rate of a data frame when the received SNR is 8 dB is 4.11×10^{-2} and that of an ACK frame is only 4.59×10^{-4}.

3.2.2. Channel estimation and rate selection

REACT can employ any of existing protocols in the literature for channel estimation and rate selection [30,31]. For the evaluation of REACT, we choose to use the following.

For the channel estimation, we used the instantaneously received SNR for the data frame preamble. For the rate selection algorithm, we used the MPDU-based adaptation algorithm in [10]. The basic idea is that the wireless station computes off-line a table of transmission modes indexed by the system status, and each entry is the best transmission mode for the purpose of maximizing the expected goodput under the corresponding system status. The system status is characterized by a triplet (l, s, n), where l is the data payload length, s is a received SNR value, and n is the frame retry count. The table is then used run-time to determine the best transmission mode for the subsequent data transmissions. More optimized techniques could be used, e.g., online calibration of SNR thresholds for the specific wireless cards as done in [8,25].

3.3. When to decrease: identification of frame losses

The key issue in rate decreasing is how to figure out the reason of frame losses that are due to channel errors or collisions. In order to cautiously differentiate frame losses, we exploit the coherence time in the wireless channel as follows. A time-domain signal may be correlated over a certain amount of time, so that the channel does not experience a significant variation for the duration of the coherence time after receiving a channel status feedback. From this, a transmitter who receives an ACK frame at the altered ACK rate can expect that the channel condition between the two stations will be stable for the duration of the coherence time after the time of ACK reception. Thus, the transmitter can figure out the reason of frame losses during that period as frame collisions instead of the reason being the bad channel condition. We call this time duration a *green channel period* during which stations do not suffer from frame losses due to the bad channel condition. Also, the notion of the green channel period can help to adaptively use RTS probing [4,7]. Note that, the degree of channel variation (or the length of the coherence time) depends on the mobility of the terminals, but it is dynamic and not easily determined in advance. Thus, we also contrive a way to dynamically estimate the green channel period. The details of the definition and estimation of the green channel period, and the enhancement of adaptive RTS probing leveraging the green channel period will be discussed in the following. If there are two consecutive frame losses when the green channel period expires, then the transmitter will switch to the next lower rate since transmission failures are assumed to be due to bad channel conditions.

3.3.1. Definition of the green channel period

Wireless channels are known to be time varying in mobile environments, i.e., the channel quality that is observed at different times will be different. However, a time-domain signal is correlated over a short time period, i.e., the channel does not experience a significant variation over a short time. This nature of the channel is typically characterized by the coherence time (T_c).

In REACT, we introduce the flexible version of the coherence time: *X% coherence time* which is defined as the time duration (or the range of Δt) over which the autocorrelation function of time (or $A_c(\Delta t)$) is greater than $\frac{X}{100}$. Assuming the Rayleigh fading channel to model the wireless channel, the normalized autocorrelation function at a constant velocity[3] can be approximated by a zeroth order Bessel function of the first kind at delay Δt:

$$A_c(\Delta t) = \frac{E\{h_b(t)h_b(t+\Delta t)^*\}}{E\{|h_b(t)|^2\}} \approx J_0(2\pi f_m \Delta t) \tag{3}$$

where $h_b(t)$ is the channel impulse response and f_m is the maximum Doppler spread. We use 50% coherence time by default; thus, the default coherence time being the range of Δt such that $A_c(\Delta t) > \frac{50}{100}$.

At last, we define the *green channel period* (T_g) as:

$$T_g = [t_{alt}, t_{alt} + T_c) \tag{4}$$

where t_{alt} is the moment when a transmitter receives an ACK frame at the altered ACK rate. In other words, the transmitter that receives an ACK frame at the altered ACK rate will assume that the channel is favorable for the duration of T_c. Thus, frame losses that occur during the green channel period are assumed to be caused by collisions, and not by the bad channel condition.

3.3.2. Estimation of the green channel period

The coherence time that is used to define the green channel period in Section 3.3.1 is calculated by assuming that the stations move at a constant velocity. In reality, the mobility of the terminals varies over time; thus, the coherence time of the channel also changes depending on the mobility. In order to estimate the coherence time depending on the varying channel environment, the threshold of the auto-correlation of the signal to estimate the coherence time is dynamically changed as follows.

We begin with 50% coherence time by default, and the percentage X is adjusted according to the following rules, while the maximum and minimum values of X are 90 and 10, respectively.

- After the green channel time expires, if data frame transmissions at the same rate are still successful, then it may imply that the coherence time has been calculated too restrictively. Thus, X is lowered by 5 to lengthen the coherence time $(X \Leftarrow X - 5)$.
- If a data frame is lost while a preceding RTS frame has been successful in the green channel period, then the channel condition must have been changed faster than expected. This presumption is based on the fact that an RTS frame that is being transmitted at a robust rate has already reserved the channel so that a data frame loss must be due to the poor channel condition (not due to collisions), though the channel was expected to be fine. In this case, X is increased by 10 to shorten the coherence time $(X \Leftarrow X + 10)$.
- If a transmitter receives an ACK frame at the altered ACK rate within the green channel period, then X is increased by 10 to shorten the coherence time. This is because the channel variation becomes faster than expected $(X \Leftarrow X + 10)$.

When we adjust the green channel period, the unit of change is conservatively chosen for lengthening the coherence time $(X \Leftarrow X - 5)$ compared with shortening it $(X \Leftarrow X + 10)$. The rationale behind this is that the green channel period should be kept as accurate as possible to avoid the false positive cases. Otherwise, the rate adaptation algorithm may make a false decision based on the erroneous channel status information.

3.3.3. Adaptive use of RTS probing

Although the reasons of frame losses can be identified within the green channel period, this cannot be achieved when the green channel period expires. When the green channel period expires, REACT needs a Supplementary mechanism such as RTS probing.

The usage of RTS probing has been accepted as a means of differentiating the frame errors that are caused by collisions from those that are caused by poor channel conditions [4,7,16]. The underlying rationale is that because of its small sizes (or short transmission times) and robust transmission rates, all RTS transmission failures are assumed to be caused by collisions, and not by channel errors. In addition, once the RTS transmission succeeds, subsequent data transmission failure comes from channel errors since the channel is already reserved by the RTS frame. Thus, enabling the RTS option before transmitting data frames can be used to identify the reason of transmission failures when some transmission failures have occurred.

Since transmitting RTS frames can add overhead, the prior solutions define a set of rules to selectively turn on or off the RTS option. For example, the CARA [4] enables the RTS option when a

---

[3] Since WLANs are typically used in low mobility environments, we set the constant velocity to 1 m/s, which is an average pedestrian walking speed [28].

data frame transmission fails, and disables it when successful data frame transmission occurs. The RRAA [7] maintains the *RTS window*, which means that the number of data frames that will be transmitted with the RTS option. The size of the RTS window is adapted to the level of collisions by observing the transmission history.

Based upon the concept of the RTS window in [7], we define a new set of rules for adapting the size of the RTS window that both exploit the channel status information obtained from the green channel period and thoroughly examines the transmission patterns. The general principle is that the RTS window is multiplicatively doubled and halved in the presence and absence of a collision, respectively. The RTS window is additively incremented by 1 if the RTS probing is needed to test the collision level, and the RTS window is unchanged otherwise.

Specifically, either a transmission failure of an RTS frame or a transmission failure of a data frame in the green channel period without the RTS option leads us to conclude that the error is caused by a collision; thus, the RTS window is doubled. When a data transmission succeeds without the RTS option, the RTS window is halved since no collision has apparently happened. If a data transmission without the RTS option fails after the green channel period expires, then we perform the RTS probing to examine the reason of frame losses; thus, the RTS window is increased by 1. When an RTS transmission succeeds, the subsequent data transmission will not be subject to a collision and hence there is no information of the current collision level; thus, the RTS window is unchanged. The adaptive use of RTS probing explained so far is summarized in Algorithm 1. Note that the RTS option is disabled or enabled depending on whether *RTScnt* is 0 or not, respectively.

Algorithm 1. Adaptive use of RTS probing

```
$RTSwnd = 0$; $RTScnt = 0$;
if $RTScnt = 0$ then {RTS option disabled}
  if DATA transmission succeeds then
    $RTSwnd \Leftarrow \lfloor RTSwnd/2 \rfloor$;
  else
    if green channel period not expires then
      $RTSwnd \Leftarrow max(2, RTSwnd * 2)$;
    else
      $RTSwnd \Leftarrow RTSwnd + 1$;
    end if
  end if
  $RTScnt \Leftarrow RTSwnd$;
else {RTS option enabled}
  if RTS transmission fails then
    $RTSwnd \Leftarrow RTSwnd * 2$; $RTScnt \Leftarrow RTSwnd$;
  else
    $RTScnt \Leftarrow RTScnt - 1$; {$RTSwnd$ is unchanged}
  end if
end if
```

4. Performance evaluation

In this section, we evaluate the performance of the REACT by using the *ns-2* simulator that is augmented to support the 802.11 OFDM PHY.[4] First, We explain the default simulation setup, and we then evaluate the performance of the REACT compared with other rate adaptation schemes as we vary the distances, number of contenders, frame sizes, mobility and channel conditions.

4.1. Simulation setup

Unless specified otherwise, our simulation experiments are configured as follows. We assume that 6, 12 and 24 Mbps, which are mandatory rates in the 802.11a specification, are in the basic rate set. We simulate the indoor office environment in which WLANs are typically used. We use a log-distance path-loss model with the path-loss exponent of four [26]. We also use the Ricean fading model with Ricean *K* factor of 3 dB to reflect the short-term multi-path fading effect in the indoor wireless channel [32]. Most of the communication parameters are from the 802.11 OFDM PHY specification [29], including power constraints, sensitivity thresholds, etc. We use LLC/IP/UDP as the upper layer protocol suite, and the MAC layer payload length is set to 1500 bytes. All stations are static and they always have a frame to transmit. Each station transmits frames for 200 s and the results of the last 100 s are taken into account to avoid the initial dynamics. We repeat this run 20 times, and we then average them.

We evaluate the following schemes in terms of the throughput (in Mbps): (1) our proposed scheme (referred to as REACT), (2) the ARF scheme (referred to as ARF), (3) the CARA scheme (referred to as CARA), (4) the RRAA schemes without or with adaptive RTS feature (referred to as RRAA and RRAA + ARTS, respectively),[5] and (5) the RBAR scheme (referred to as RBAR).

4.2. Transmission rate adaptation over time

We begin with a simple experiment that traces the rate adaptation decisions of the testing schemes over time in order to illustrate why the above schemes show different performance. This evaluation is done in the simple one-to-one topology in which a station continuously transmits frames to the other station. Both stations are static and the distance between the two stations is arbitrarily set to 30 m.

Given the same time-varying channel quality variation that is generated by the default simulation setting, the different behaviors of ARF, RRAA, RBAR, and REACT are shown in Fig. 2(a)–(d), respectively.[6] The *x*-axis represents the simulation time span from 1.75 to 2.0 s. The left-hand *y*-axis represents the data transmission rates chosen by each testing scheme at the particular time. The right-hand *y*-axis represents the signal-to-noise ratio (SNR) at the receiver in dB. Note that the total number of transmission attempts for each scheme is different for the same simulation time of 0.25 s. This implies that each scheme differently switches its transmission rates that result in different throughput. In these figures, the following legends are used:

- *Sinusoidal curve*: A variation of the SNR at the receiver.
- *Circle point*: A successful data frame transmission.
- *Cross point*: An erroneous data frame transmission.

We intentionally exclude the transmission attempts of the RTS frames and the response frames (CTS or ACK) for simplicity.

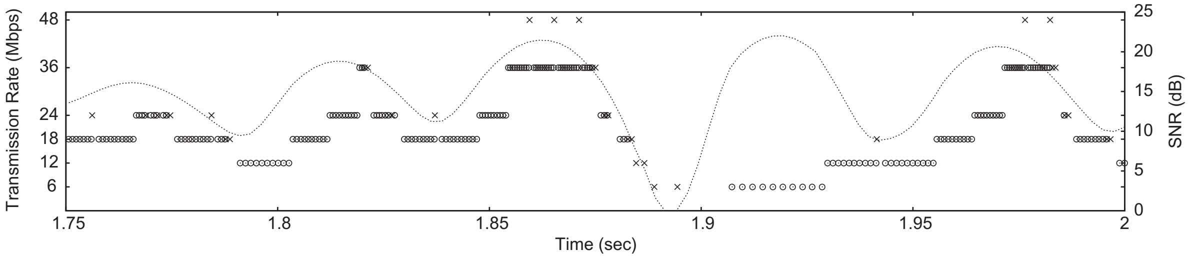
The ARF results. Fig. 2(a) shows how the ARF scheme reacts to the channel variation. We observe some inefficiencies of the ARF's

---

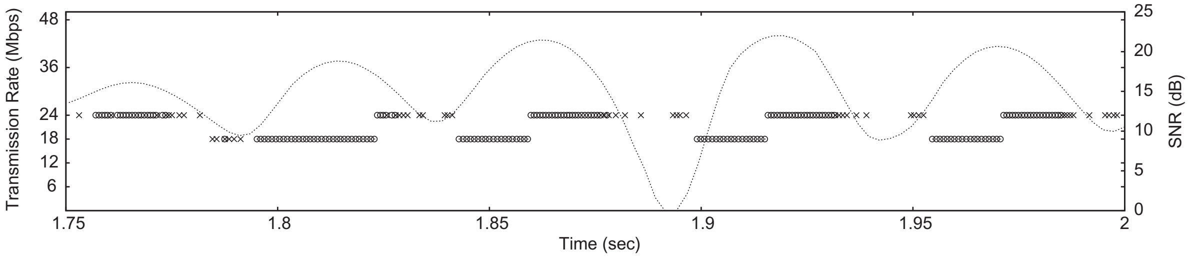
[4] We exclude 9 Mbps rate in 802.11a from the set of candidate rates and simulation since it has been known to always perform worse than 12 Mbps under all SNR conditions [10].

[5] We evaluate RRAA apart from RRAA + ARTS since RRAA exhibits the characteristics of the open-loop rate adaptation schemes that estimate the channel condition by calculating a frame loss ratio for recent transmissions, and shows substantially different performance in certain scenarios.

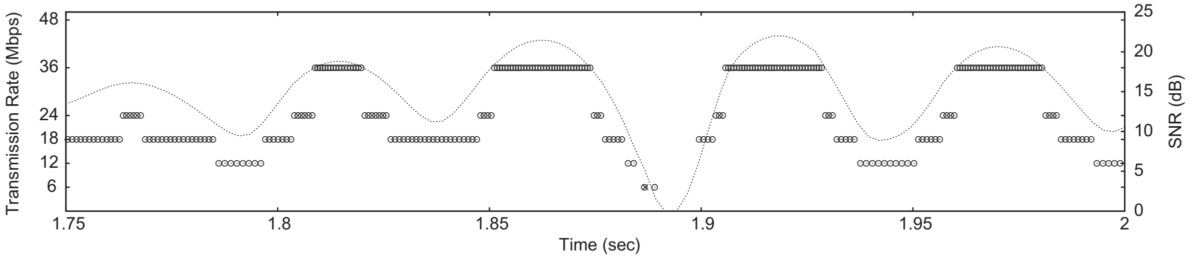
[6] We omit the traces of CARA and RRAA + ARTS due to the space limit since their operations are basically similar with ARF and RRAA respectively when there is no contending station, except that they exploit RTS probing adaptively when data transmission failures occur.



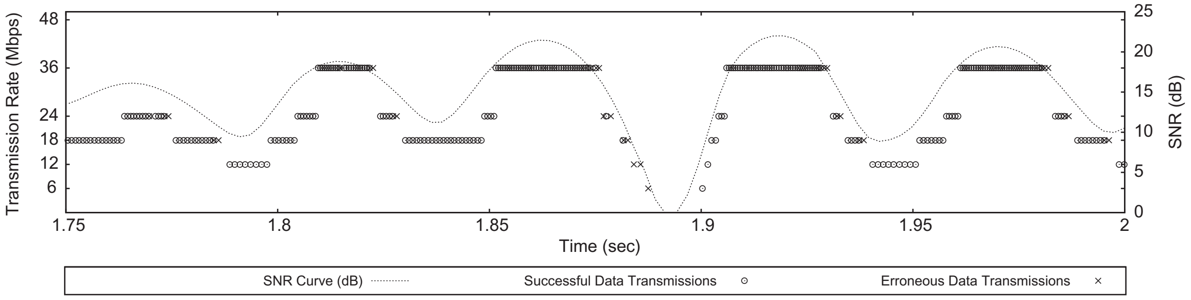
(a) ARF



(b) RRAA



(c) RBAR



(d) REACT

Fig. 2. Adaptability comparison of our proposed scheme (REACT) against ARF, RRAA, and RBAR.

operation. First, 10 consecutive successful transmissions that are necessary for switching to the next higher bit rate are too conservative to react to improving signal quality. In particular, 10 consecutive data transmissions at low bit rates consume much longer time than those at high bit rates. For example, ARF cannot transmit at high transmission rates at the top of the fourth concave part of the SNR curve, which appears after 1.9 s. Fortunately, two consecutive transmission errors that are necessary for switching to the next lower rate are responsive enough to react to the deteriorating channel conditions. Second, the ARF scheme will superfluously attempt to transmit a data frame at the next higher rate for channel probing after 10 consecutive transmission successes even though the channel quality is not improved, which incurs non-negligible throughput reduction. This is inevitable because its channel estimation and adaptation rely on the counters. Unfortunately, these two inefficient aspects (i.e., slow adaptation and unnecessary probing) cannot be concurrently solved as long as the counter is exploited for estimating the channel. If the count for rate-increase attempt is reduced for achieving fast responsiveness (e.g., the number of consecutive successes becomes lower than 10), then more wireless link bandwidth will be wasted for probing.

The RRAA results. Fig. 2(b) shows that the rate decision of RRAA cannot closely catch up with the change of the SNR curve. The RRAA scheme slowly reacts to the channel because of its large channel estimation window (e.g., 6–40 frames in the original proposal [7]). Despite its optimization technique for improving its responsiveness to channel fluctuation, which calculates best/worst possible loss rate and compares it against threshold values before finishing each estimation window, the number of transmissions that needed for switching the rates is still much larger than those of the competing schemes. Note that, as observed in the ARF, the number of data transmissions that is required for the rate-switching decision should be kept small in order to exploit the transiently improved channel condition. However, RRAA is essentially unable to keep the estimation window small since the estimation windows size should be larger than the inverse of the threshold value (if the threshold value is 10%, then the estimation window should be larger than or equal to 10).

The RBAR results. Fig. 2(c) represents how critical the feedback from the receiver is for making the accurate transmission rate decision. RBAR accurately and timely switches the transmission rate for the subsequent data frame corresponding to the channel variation by using the channel status information conveyed via modified CTS frames. However, it should be noted that RBAR obtains its accuracy and responsiveness at the cost of the RTS and CTS frames' exchanges that consumes precious link bandwidth. For example, in Fig. 2(c), RBAR switches its transmission rate only 26 times while it is exchanging the RTS/CTS frames 277 times; this means that around 9 out of 10 RTS/CTS frame exchanges may be wasteful. What is worse is that exchanging RTS/CTS frames relatively adds more overhead when high transmission rates are being used for data frame transmissions [24].

The REACT results. Fig. 2(d) shows that REACT effectively reacts to the channel variations using opportunistic ACK transmission rate decision. When the channel condition becomes good enough to use the next higher bit rate (i.e., at the uphill SNR curve), the receiver opportunistically informs the transmitter of this situation by using the altered ACK rate. In addition, REACT can quickly switch to the next lower rate as ARF does when the channel condition becomes deteriorated (i.e., at the downhill SNR curve). The explicit feedback enables the transmitter to react to the channel dynamics faster than ARF and RRAA when compared against ARF and RRAA, which require a number of data transmissions to figure out the improved channel condition. Also, compared against the RBAR scheme, the receiver in the REACT sends feedback *only once* to increase the bit rate, and thus inducing the marginal overhead.

Summary. Table 2 compares the testing schemes under the scenario in Fig. 2 in terms of the number of transmission attempts, the number of successful/erroneous frame transmissions, and the corresponding throughput for the observed time period. The results confirm that REACT performs better than the other schemes due to its efficient feedback mechanism. RBAR performs better than ARF and RRAA since it experiences fewer transmission errors due to the accurate feedback information. However, its performance gain is reduced by its control frame overhead.

4.3. Results with various distances

We then evaluate the tested schemes with various distances. Since the distance is a major factor of radio signal attenuation, we can observe the different behaviors of the testing schemes depending on various conditions. A static station continuously transmits frames to the other station, and the distance between the two stations is varied between 5 and 60 m. Fig. 3 shows the average throughput of the tested schemes: ARF, CARA, RRAA, RRAA + ARTS, RBAR, single-rate schemes, and REACT. In general, the throughput of all the schemes decreases as the distance between the two stations increases, and the throughput curves of the rate adaptation schemes roughly follow the outer envelope of those of the single-rate schemes. We have three notable observations as follows.

Table 2
Comparison of four testing schemes.

	ARF	RRAA	RBAR	REACT
# of tx attempts	266	241	277	323
# of tx successes	231	186	275	289
36 Mbps	58	0	122	148
24 Mbps	50	96	40	44
18 Mbps	81	90	86	76
12 Mbps	32	0	25	20
6 Mbps	10	0	2	1
Throughput (Mbps)	11.28	9.08	13.43	14.11

First, not only does REACT achieve better throughput than the other schemes over the entire range, but also its throughput curve is above the best throughput curve combined from the single-rate schemes. When the channel temporarily becomes better or worse due to fading effects, the fast-responsive rate adaptation can achieve higher throughput than the best achievable throughput of single-rate schemes around the boundary of the best performance regions of two single-rate schemes. This is because the single-rate scheme cannot exploit the temporary channel dynamics. For example, suppose that the channel condition goes poor and hence accommodates at bit rate of 36 Mbps, the single-rate scheme with 48 Mbps cannot successfully send packets. On the other hand, if the channel condition gets better and hence accommodates up to 48 Mbps, the single-rate scheme with 36 Mbps cannot fully utilize the channel condition. REACT, however, can fully exploit the transient channel dynamics by quickly switching to the optimal rate. This highlights how critical the responsive reaction to the channel variation is for rate adaptation; it should seek to switch to the optimal rate to opportunistically exploit the transient channel dynamics.

Second, RBAR, with using accurate feedback, performs even worse than ARF-based schemes (i.e., ARF and CARA) when two communicating nodes are located either closely to each other or near the boundary of the communication range. It is because the advantage of the accurate feedback is reduced when the given channel status practically allows few bit rates. In particular, the delivery of the feedback information may incur the substantial overhead when a station can use high transmission rates. As the data rate increases, the control packets, which are transmitted at

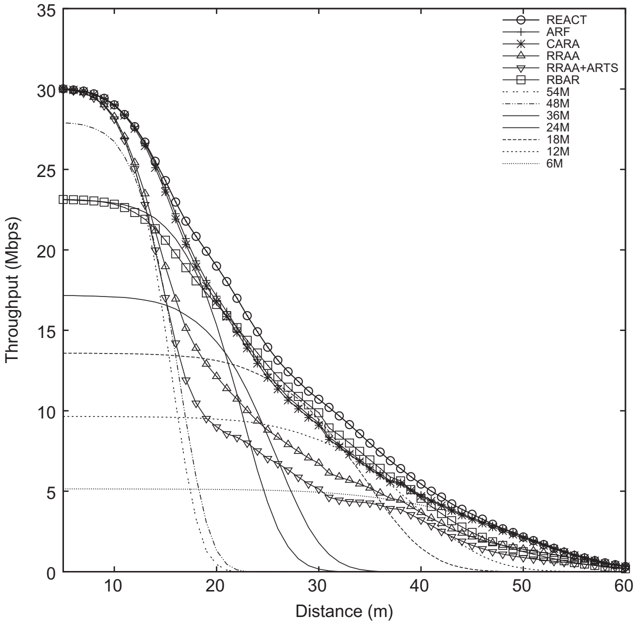


Fig. 3. Throughput comparison of our proposed scheme (REACT) against ARF, CARA, RRAA, RRAA + ARTS, RBAR, and single-rate schemes for one-to-one topology networks with various distances.

one of the rates in the basic rate set, continue to waste proportionally more and more transmission time [24].

Third, in accordance with Section 4.2, the RRAA family (RARA and RARA + ARTS) performs the worst among all the tested schemes. First of all, the large size of the estimation windows in the RRAA family cannot make use of the short-term channel link variations. In addition, the large size of the estimation windows results a side effect that causes a number of transmission failures to trigger unnecessary RTS probings in the case of RRAA + ARTS. Thus, the throughput of RRAA + ARTS is worse than that of RRAA over most of the ranges.

4.4. Results with varying number of contending stations

To study how efficiently REACT operates in collision-prone environments, we now evaluate the performance of the tested schemes in a star topology. In this topology, a varying number of contending stations are equi-distantly placed on a circle centered around an access point with a 5 m radius. It should be noted that at a 5 m distance, a station can always transmit data frames at 54 Mbps, as shown in Fig. 3; thus, stations that are located at this distance should not lower their data rate.

Simulation results are plotted in Fig. 4. In general, the REACT scheme achieves the higher performance than the others regardless of the number of contending stations. ARF and RRAA perform considerably poorer than the other schemes because they cannot figure out that the cause of transmission errors is collisions. In addition, RBAR almost achieves constant throughput regardless of the number of contending stations.

The un-scalable behaviors of ARF and RRAA are explained by two aspects. First, both ARF and RRAA cannot differentiate between the collisions and the channel errors, and thus unnecessarily decrease the transmission rate. Second, when the data rate is mistakenly lowered, both ARF and RRAA cannot easily recover from its change. For instance, 10 consecutive successful transmissions for ARF and the small estimated loss ratio for RRAA are not likely to happen as the number of contending stations increases. Note that RRAA performs better than the ARF in collision-prone environments because lowering the data rate from two successive failures in ARF causes the stations to lower their bit rates faster than does RRAA.

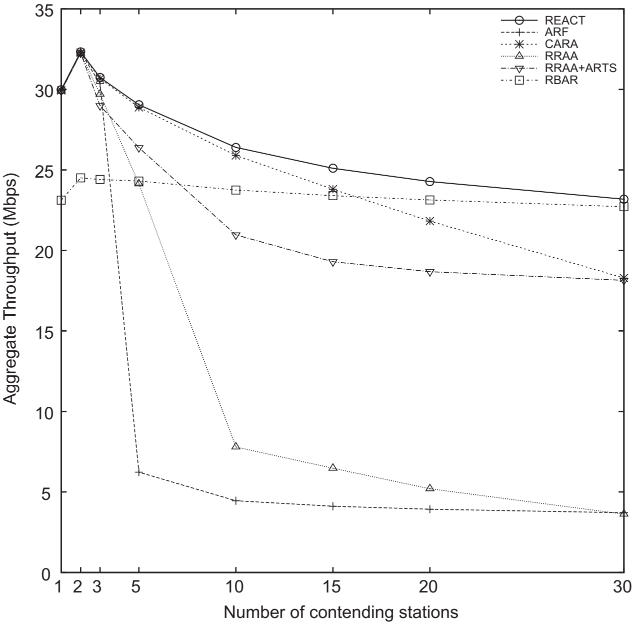


Fig. 4. Throughput comparison of our proposed scheme (REACT) against ARF, CARA, RRAA, RRAA + ARTS, RBAR, with varying number of contending stations.

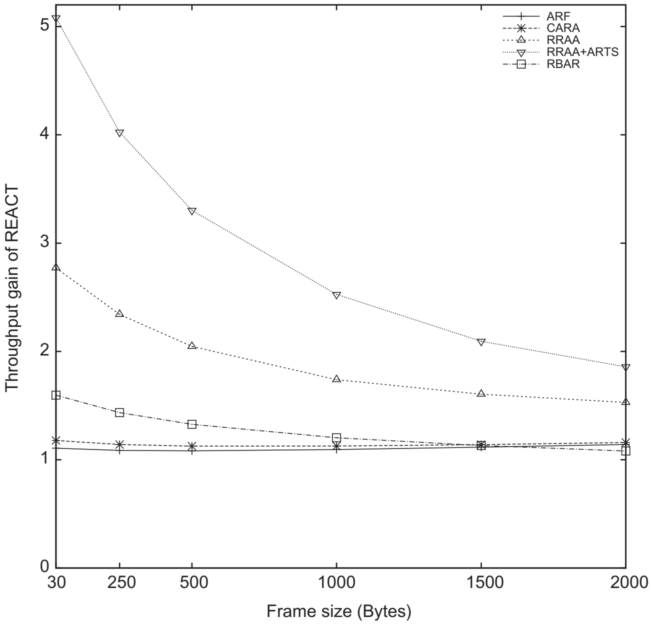


Fig. 5. Throughput comparison of our proposed scheme (REACT) against ARF, CARA, RRAA, RRAA + ARTS, RBAR for one-to-one topology with various frame sizes.

The CARA and RRAA + ARTS schemes perform better than the ARF and RRAA schemes due to the CARA and RRAA + ARTS schemes' abilities to identify the reason of transmission errors. After experiencing transmission failures, the CARA and RRAA + ARTS schemes adaptively use RTS probing according to their own algorithms. CARA generally performs better than RRAA + ARTS because CARA can switch to the next higher rate faster than can RRAA + ARTS. Under heavy-collision environments, the performance of both CARA and RRAA + ARTS schemes is degraded because severe collisions causes the stations to mistakenly decide the rate-decrease in spite of using the RTS probing mechanism.

Finally, REACT outperforms the other schemes regardless of the number of contending stations because of two reasons. First, the use of adaptive RTS probing that exploits the green channel period can allow stations to effectively differentiate the collisions from the channel errors. Second, the REACT can quickly switch to the optimal transmission rate via the opportunistic feedback.

RBAR in which the RTS/CTS frames are always exchanged before each data frame, results in poor throughput due to the excessive overhead especially when the number of contenders is small. However, the advantage of channel reservation by the short RTS/CTS frames reduces its overhead as the number of contending stations becomes large.

4.5. Results with various frame sizes

To investigate the impact of the frames' sizes on each testing scheme, we measure the throughput over the one-to-one topology by varying the size of data frames (30, 250, 500, 1000, 1500 and 2000 bytes). The distance between two static stations is fixed to 20 m.[7] Note that, the ratio of MAC overhead (i.e., idle time for channel contention and transmission time for RTS/CTS frames) to

---

[7] Three single-rate schemes, i.e., 18, 24 and 36 Mbps, achieve similar throughput around the distance of 20 m. Thus, we can focus on the efficacy of rate adaptation schemes by observing how they switch rates depending on the varying channel condition at 20 m.

the duration of a successful data frame transmission increases in inverse proportion to the size of a data frame.

The throughput gain of REACT over each testing scheme is plotted in Fig. 5. Overall, the REACT performs better than the other schemes regardless of the frame size, i.e., the throughput gain of REACT is always above 1. Specifically, we have the following observations. First, ARF, CARA and RBAR schemes achieve a relatively fair performance in terms of throughput. In particular, the throughput gain of REACT over RBAR increases as the size of the data frame decreases, while those of REACT over ARF and CARA are not noticeably changed. This is because the exchange of the RTS/CTS frames for feedbacks in RBAR consumes proportionally more and more time as the size of data frames decreases. Second, the RRAA family achieves the worst performance among all the testing schemes due to their inaccurate rate decisions. As observed in Table 2, the RARA family has a higher transmission failure ratio than the others, and RARA family's performance continues to be more degraded as the sizes of data frames are decreased. That is, the small frame size (i.e., the short data transmission time) creates more chances of data transmission attempts for the simulation time, so that the RRAA family results in the larger number of transmission failures. In addition, a transmission failure will lengthen the idle time due to the exponential backoff as well.

4.6. Results with Varying Ricean Parameter K

Now, we evaluate the performance of the testing schemes under various Ricean fading parameter *K*, which represents the relative strength of the line-of-sight (LOS) component of the received signal. In the Ricean distribution, parameter *K* is defined as the ratio of the power of the LOS component to the power of the other (non-LOS) multipath components. In terms of dB, it is given by

$$K(\mathrm{dB}) = 10\log\frac{A^2}{2\sigma^2} \tag{5}$$

where A denotes the peak amplitude of the LOS signal, and $2\sigma^2$ is the average power of the non-LOS multipath components [33]. For $K = 0$, the Ricean distribution becomes equivalent to the Rayleigh distribution, in which the channel has no LOS component; only the reflected signal components are received and the overall channel quality is degraded. As *K* increases, the LOS component becomes dominant and hence the overall signal quality is improved.

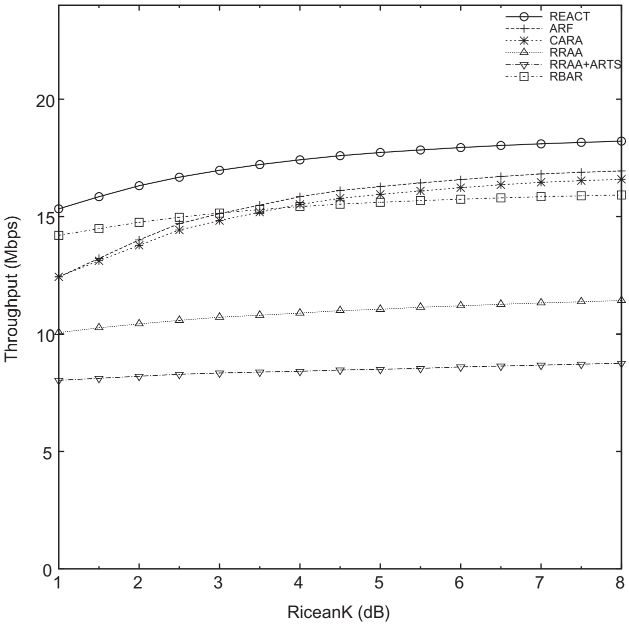


Fig. 6. Throughput comparison of our proposed scheme (REACT) against ARF, CARA, RRAA, RRAA + ARTS, RBAR for one-to-one topology as a function of Ricean parameter *K*.

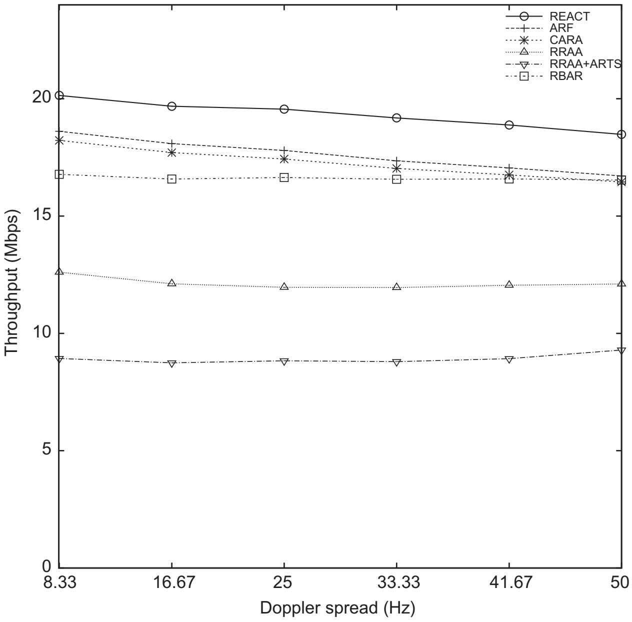


Fig. 7. Throughput comparison of our proposed scheme (REACT) against ARF, CARA, RRAA, RRAA + ARTS, RBAR for one-to-one topology as a function of Doppler spread (Hz).

Fig. 6 shows the throughput of each scheme in the one-to-one topology versus the Ricean parameter *K*. The distance between two stations is fixed to 20 m. In general, the throughput of each scheme is improved as the parameter *K* increases because they exploit the improved channel condition as *K* increases. Specifically we have the following observations. First, REACT outperforms the others in terms of the throughput for the entire range of *K*. This indicates that REACT can maintain relatively stable performance not only in good channel conditions but also in poor channel conditions. Second, ARF, CARA and RBAR achieve relatively poor throughput than REACT, and the RRAA family exhibits the lowest throughput. In particular, the throughput of RBAR is higher than those of ARF and CARA when the channel condition is bad; the situation is reversed as the parameter *K* increases. This indicates that the accurate channel information is more valuable when the channel condition becomes worse. On the other hand, both ARF and CARA can operate fairly well when the channel is almost deterministic.

4.7. Results with various Doppler spread

In order to study the effect of indoor mobility, we compare the performance of the tested schemes on the various mobility conditions, i.e., different Doppler spread values in the Ricean model. We vary the values of maximum Doppler spread from 8.33 to 50 Hz (which corresponds to the mobile speeds from 0.5 to 3 m/s), while the distance between two stations is fixed to 20 m.[8] As shown in Fig. 7, REACT achieves the best throughput over the entire range of mobility conditions.

---

[8] Although the distance between two stations is fixed, the channel between them still shows the time-varying nature because of the Ricean channel model. In addition, the different Doppler spread values used in the Ricean model describes the relative motion between the transmitter and receiver.

Among the open-loop schemes, ARF and CARA show a different tendency from the RRAA family depending on the varying mobility. The performance of the former schemes is degraded as the mobility increases, but that of the latter schemes is almost not affected. At a first glance, these results are counter-intuitive because both ARF and CARA schemes can more quickly react to the channel variation than the RRAA family as discussed in Section 4.2. However, the adaptivity of ARF and CARA schemes cannot meet the channel dynamics as the mobility increases; i.e., the trials to exploit transient channel dynamics result in more frequent transmission failures. On the other hand, the RRAA family essentially cannot exploit the transient variation of the channel dynamics; so they are not hindered by the fluctuation of the wireless channel link.

RBAR achieves less throughput than ARF and CARA when the Doppler spread is less than or equal to 41.67 Hz (which corresponds to the mobile speed of 2.5 m/s). As the mobility increases, the throughput of RBAR becomes close to that of the ARF and CARA schemes since its throughput is least affected by the varying mobility. This is because the stations that are using RBAR convey the channel information via the RTS/CTS frames so that the prompt channel information for subsequent data frames is available to the transmitters.

4.8. Adaptation of green channel period

As stated in Section 3.3.2, the channel coherence time is calculated by assuming that the stations move at a constant velocity. The stations, however, may change its moving speed dynamically. To quantitatively study the usefulness of proper estimation of green channel period, we compare REACT with its variants with the fixed channel coherence time. Table 3 lists the normalized throughput of testing schemes (relative to the best scheme in each row). The number of contending stations and the maximum mobile speed (i.e., maximum Doppler spread in the Ricean channel) are denoted by n and v, respectively, and $X = x$ stands for the REACT variant which defines the fixed green channel period with x% channel coherence time (which is defined in Section 3.3.2).

First, it is important to note that no certain REACT variant with fixed green channel period can consistently achieve the best throughput. For example, $X = 10$ is the best scheme when n is 10 and v is 0.5 whereas it becomes the worst when n is 1 and v is 3. REACT, however, achieves stable performance thanks to its dynamic channel coherence time estimation. Second, we observe that the REACT variant with shorter green channel period achieves better throughput as the velocity increases when there is no contention. This is due to the fact that too large green channel period may misinterpret the reason of frame losses within the green channel period as collisions; thus preventing responsive rate-fallback. Third, when there are 10 contending stations, the disadvantage of short green channel period is alleviated because too short green channel period can misidentify the reason of frame losses due to collisions as the deteriorated channel condition.

Table 3
Effect of green channel period estimation.

n	v	REACT	$X = 10$	$X = 50$	$X = 90$
1	0.5	0.9998	0.9997	1.0000	0.9992
	1	0.9992	0.9964	0.9975	1.0000
	2	0.9995	0.9786	0.9881	1.0000
	3	0.9992	0.8476	0.8960	1.0000
10	0.5	0.9723	1.0000	0.9923	0.9641
	1	0.9972	0.9982	1.0000	0.9959
	2	0.9999	0.9828	0.9924	1.0000
	3	0.9957	0.9627	0.9777	1.0000

4.9. Results with random topologies and configurations

To verify the robustness of REACT, we consider more complex random topologies and configurations by varying the previous factors together. With varying number of contending stations in star topology, we simulate 50 random scenarios as follows. For each scenario, stations are uniformly distributed in a circular arena within the diameter of 50 m. Each station chooses a random frame size in the range of 30 to 2000 bytes, and transmits frames to the access point located at the center of the arena. Each scenario is also configured with different channel conditions: Ricean parameter K is randomly selected in the range of 1–8 dB, and Doppler spread ranges from 5 to 50 Hz. Fig. 8 shows the aggregate throughput of each scheme by varying number of contending stations. It is observed that REACT robustly outperforms the others in the complex scenarios.

5. Conclusion

In this paper, we proposed a new rate adaptation scheme that is termed as *Rate Adaptation using Coherence Time* (*REACT*). REACT has two main features. First, a transmitter can responsively increase the transmission rate corresponding to the improving channel condition by receiving the feedback information via an ACK frame from the receiver at an altered ACK rate. This feedback mechanism can be implemented at the device driver level and it induces a negligible extra overhead. Second, the reason of frame losses is effectively identified by exploiting the fact that the channel condition will be favorable for the current transmission rate during the channel coherence time. Moreover, the information of the channel coherence time can further be exploited to enhance the adaptive use of RTS probing, which differentiates the channel-error-induced frame losses from the collision-induced ones.

We carried out comprehensive *ns-2* simulation experiments that consider indoor environments. Our simulation scenarios include varying the distance, number of contending stations, frame size, mobility and channel conditions. We found that REACT

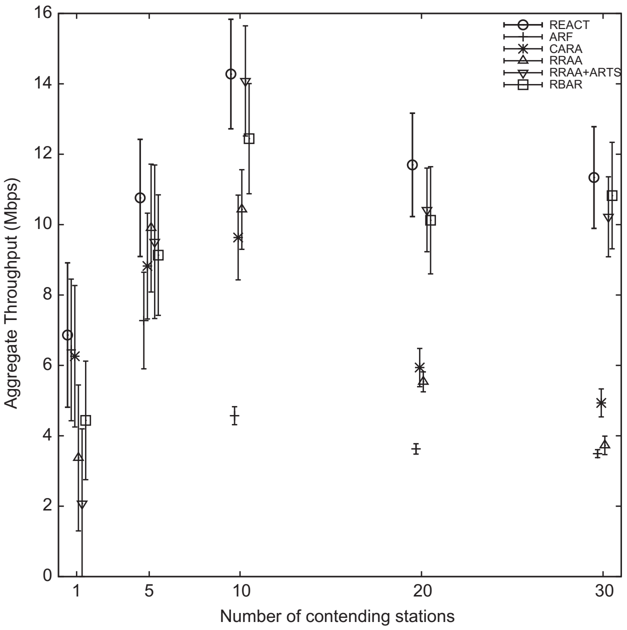


Fig. 8. Throughput comparison of our proposed scheme (REACT) against ARF, CARA, RRAA, RRAA + ARTS, RBAR for random topologies and configurations, with varying number of contending stations. The vertical bars indicate 95% confidence intervals.

consistently obtains higher throughput than the other testing schemes: ARF, CARA, RRAA and RBAR.

In the future, we plan to conduct further research to enhance REACT as follows. First, the current version of REACT only considers the altered ACK rate as a mean to signal the improved channel condition. The receiver in REACT, however, could inform the transmitter of the degraded channel condition using higher ACK transmission rate, so that the transmitter decreases the data rate for subsequent data frames. Second, we plan to implement a prototype of REACT using the real WLAN devices, e.g., MADWIFI [34]. Finally, the current REACT mechanism is designed for single-input–single-output (SISO) wireless networks. We plan to extend REACT for multiple-input–multiple-output (MIMO)-based 802.11n wireless networks. We expect REACT's operation in 802.11n to be viable because the 802.11n amendment [35] supports not only the backward compatibility with the original MAC but also the formal use of an alternative bit rate for control frames.

Acknowledgements

This work was supported by the National Research Foundation of Korea Grant funded by the Korean Government MEST, Basic Research Promotion Fund (NRF-2010-013-D00063) and IT R&D program of MKE/KEIT [10035245: Study on Architecture of Future Internet to Support Mobile Environments and Network Diversity]. The ICT at Seoul National University provides research facilities for this study. Prof. Kwon was on sabbatical leave at Rutgers University, where he had in-depth discussions with Prof. Dipankar Raychaudhuri.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.comcom.2011.01.011.

References

[1] A. Kamerman, L. Monteban, WaveLAN-II: a high-performance wireless LAN for the unlicensed band, Bell Labs Technical Journal 2 (3) (1997) 118–133.
[2] M. Lacage, M.H. Manshaei, T. Turletti, IEEE 802.11 rate adaptation: a practical approach, in: MSWiM '04: Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems, 2004, pp. 126–134.
[3] P. Chevillat, J. Jelitto, A. Barreto, H. Truong, A dynamic link adaptation algorithm for IEEE 802.11 a wireless lans, in: IEEE International Conference on Communications 2003, ICC '03, vol. 2, 2003, pp. 1141–1145.
[4] J. Kim, S. Kim, S. Choi, D. Qiao, Cara: collision-aware rate adaptation for IEEE 802.11 WLANS, 2006, pp. 1–11.
[5] J. Choi, J. Na, K. Park, C. kwon Kim, Adaptive optimization of rate adaptation algorithms in multi-rate WLANS, in: Network IEEE International Conference on Protocols, 2007, ICNP 2007, 2007, pp. 144–153.
[6] J. Bicket, Bit-rate selection in wireless networks, Ph.D. Thesis, Massachusetts Institute of Technology, 2005.
[7] S.H.Y. Wong, H. Yang, S. Lu, V. Bharghavan, Robust rate adaptation for 802.11 wireless networks, in: MobiCom '06: Proceedings of the 12th Annual International Conference on Mobile Computing and Networking, 2006, pp. 146–157.
[8] G. Judd, X. Wang, P. Steenkiste, Efficient channel-aware rate adaptation in dynamic environments, in: MobiSys '08: Proceeding of the 6th International Conference on Mobile Systems, Applications, and Services, 2008, pp. 118–131.
[9] J. Pavon, S. Choi, Link adaptation strategy for IEEE 802.11 WLAN via received signal strength measurement, in: IEEE International Conference on Communications 2003, ICC '03, vol. 2, 2003, pp. 1108–1113.
[10] D. Qiao, S. Choi, K.G. Shin, Goodput analysis and link adaptation for IEEE 802.11a wireless lans, IEEE Transactions on Mobile Computing 1 (4) (2002) 278–292.
[11] J. Zhang, K. Tan, J. Zhao, H. Wu, Y. Zhang, A practical snr-guided rate adaptation, in: The 27th Conference on Computer Communications, INFOCOM 2008, IEEE, 2008, pp. 2083–2091.
[12] L. Verma, S. Kim, S. Choi, S.-J. Lee, Agile rate control for IEEE 802.11 networks, in: FGIT '09: Proceedings of the 1st International Conference on Future Generation Information Technology, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 237–245.
[13] G. Holland, N. Vaidya, P. Bahl, A rate-adaptive mac protocol for multi-hop wireless networks, in: MobiCom '01: Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, 2001, pp. 236–251.
[14] B. Sadeghi, V. Kanodia, A. Sabharwal, E. Knightly, Oar: an opportunistic auto-rate media access protocol for ad hoc networks, Wireless Networks 11 (1–2) (2005) 39–53.
[15] M. Vutukuru, H. Balakrishnan, K. Jamieson, Cross-layer wireless bit rate adaptation, SIGCOMM Computer Communcation Review 39 (4) (2009) 3–14.
[16] C. Hoffmann, H. Manshaei, T. Turletti, Clara: closed-loop adaptive rate allocation for IEEE 802.11 wireless lans, in: 2005 International Conference on Wireless Networks, Communications and Mobile Computing, vol. 1, 2005, pp. 668–673.
[17] Q. Pang, V. Leung, S. Liew, A rate adaptation algorithm for IEEE 802.11 WLANS based on mac-layer loss differentiation, in: 2nd International Conference on Broadband Networks, 2005, BroadNets 2005, vol. 1, 2005, pp. 659–667.
[18] H. Jung, K. Cho, Y. Seok, T. Kwo, Y. Choi, RARA: rate adaptation using rate-adaptive acknowledgment for IEEE 802.11 WLANs, in: 5th IEEE Consumer Communications and Networking Conference, 2008, CCNC 2008, 2008, pp. 62–66.
[19] P. Acharya, A. Sharma, E. Belding, K. Almeroth, K. Papagiannaki, Congestion-aware rate adaptation in wireless networks: a measurement-driven approach, in: 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks 2008, SECON '08, 2008, pp. 1–9.
[20] S. Choi, K. Park, C.-k. Kim, On the performance characteristics of WLANS: revisited, SIGMETRICS Performance Evaluation Review 33 (1) (2005) 97–108.
[21] D. Kotz, C. Newport, R.S. Gray, J. Liu, Y. Yuan, C. Elliott, Experimental evaluation of wireless simulation assumptions, in: MSWiM '04: Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems, 2004, pp. 78–82.
[22] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, S. Wicker, Complex behavior at scale: an experimental study of low-power wireless sensor networks, Tech. Rep., UCLA/CSD-TR 02-0013, UCLA Computer Science, 2002.
[23] K. Papagiannaki, M. Yarvis, W.S. Conner, Experimental characterization of home wireless networks and design implications, in: INFOCOM 2006, Proceedings of the 25th IEEE International Conference on Computer Communications, 2006, pp. 1–13.
[24] I. Tinnirello, S. Choi, Y. Kim, Revisit of RTS/CTS exchange in high-speed IEEE 802.11 networks, 2005, pp. 240–248.
[25] J. Camp, E. Knightly, Modulation rate adaptation in urban and vehicular environments: cross-layer implementation and experimental evaluation, in: MobiCom '08: Proceedings of the 14th ACM International Conference on Mobile Computing and Networking, 2008, pp. 315–326.
[26] T.S. Rappaport, Wireless Communications: Principles and Practice, Prentice Hall, NJ, 2002.
[27] G. Bianchi, Performance analysis of the ieee 802.11 distributed coordination function, Selected Areas in Communications, IEEE Journal on 18 (3) (2000) 535–547.
[28] R. Knoblauch, M. Pietrucha, M. Nitzburg, Field studies of pedestrian walking speed and start-up time, Transportation Research Record: Journal of the Transportation Research Board 1538 (1) (1996) 27–38.
[29] IEEE standard for information technology – telecommunications and information exchange between systems – local and metropolitan area networks-specific requirements – Part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications, IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999) (2007) C1–1184.
[30] C. Tepedelenlioglu, A. Abdi, G.B. Giannakis, M. Kaveh, Estimation of doppler spread and signal strength in mobile communications with applications to handoff and adaptive transmission, Wireless Communications and Mobile Computing 1 (2) (2001) 221–242.
[31] A. Goldsmith, S.-G. Chua, Adaptive coded modulation for fading channels, IEEE Transactions on Communications 46 (5) (1998) 595–602.
[32] M. Carroll, T. Wysocki, Fading characteristics for indoor wireless channels at 5 GHz unlicensed bands, 2003, pp. 102–105.
[33] A. Goldsmith, Wireless Communications, Cambridge University Press, New York, NY, USA, 2005.
[34] Multiband atheros driver for wifi (madwifi), <http://sourceforge.net/projects/madwifi/>.
[35] IEEE standard for information technology – telecommunications and information exchange between systems – local and metropolitan area networks-specific requirements – Part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 5: Enhancements for higher throughput, IEEE Std 802.11n-2009 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, and IEEE Std 802.11w-2009) (2009) c1–502.