CARA: Collision-Aware Rate Adaptation for IEEE 802.11 WLANs

[†]Jongseok Kim [†]Seongkwan Kim [‡]Sunghyun Choi
School of Electrical Engineering and INMC
Seoul National University
Seoul, 151-744, Korea
[†]{jskim,skim}@mwnl.snu.ac.kr [‡]schoi@snu.ac.kr

Daji Qiao
Department of Electrical and Computer Engineering
Iowa State University
Ames, IA 50011
daji@iastate.edu

***Abstract*— Today's IEEE 802.11 WLANs (Wireless LANs) provide multiple transmission rates so that different rates can be exploited in an adaptive manner depending on the underlying channel condition in order to maximize the system performance. Many rate adaptation schemes have been proposed so far while most (if not all) of the commercial devices implement a simple open-loop rate adaptation scheme (i.e., without feedback from the receiver), called *ARF (Automatic Rate Fallback)* due to its simplicity. A key problem with such open-loop rate adaptation schemes is that they do not consider the collision effect, and hence, malfunction severely when many transmission failures are due to collisions. In this paper, we propose a novel rate-adaptation scheme, called *CARA (Collision-Aware Rate Adaptation)*. The key idea of CARA is that the transmitter station combines adaptively the Request-to-Send/Clear-to-Send (RTS/CTS) exchange with the Clear Channel Assessment (CCA) functionality to differentiate frame collisions from frame transmission failures caused by channel errors. Therefore, compared with other open-loop rate-adaptation schemes, CARA is more likely to make the correct rate adaptation decisions. Through extensive simulation runs, we evaluate our proposed scheme to show that our scheme yields significantly higher throughput performance than the existing schemes**

I. INTRODUCTION

During the last decade, IEEE 802.11 Wireless LAN (WLAN) has been widely accepted as the dominant technology for (indoor) broadband wireless networking. The 802.11 standard defines Medium Access Control (MAC) layer and Physical layer (PHY) specifications [1]. The mandatory contention-based channel access function is called the Distributed Coordination Function (DCF), which is based on Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA). The optional polling-based Point Coordination Function (PCF) is also specified in the standard, but it is rarely implemented in today's 802.11-compliant products. By employing different modulation and channel coding schemes, the 802.11 PHYs provide multiple transmission rates. For example, the original 802.11 standard specifies three low-speed PHYs operating at 1 and 2 Mbits/s (Mbps), and three high-speed PHYs are additionally defined as supplements to the original standard: (1) the 802.11b PHY [2] supporting four transmission rates up to 11 Mbps at the 2.4 GHz band, (2) the 802.11a PHY [3] supporting eight transmission rates up to 54 Mbps at the 5 GHz band, and (3) the 802.11g PHY [4] supporting 12 transmission rates up to 54 Mbps at the 2.4 GHz band.

Since the 802.11 standard does not specify any algorithm and/or protocol to efficiently utilize the multiple transmission rates, many rate adaptation schemes have been proposed [11]–[13], [15], [18]–[21]. The effectiveness of a rate adaptation scheme depends on how fast it can respond to the variation of wireless channel. In addition, in a multi-user environment where frame collisions are inevitable due to the contention nature of the 802.11 DCF, the effectiveness of a rate adaptation scheme also depends greatly on how the collisions may be detected and handled properly.

Unfortunately, most open-loop rate-adaptation schemes do not consider the collision effect, and hence, may malfunction severely when many transmission failures are due to collisions. For example, the widely-adopted ARF (Automatic Rate Fallback) scheme [15] does not work properly in multi-user environments since it decreases the transmission rate upon consecutive frame collisions, as presented in [10] based on both simulation and empirical results. In contrast, the collision effect fades in closed-loop rate-adaptation schemes, such as RBAR [13] and OAR [18], thanks to the interaction between the transmitter and the receiver.

Based on the above observation, we propose a novel rate-adaptation scheme, called *CARA (Collision-Aware Rate Adaptation)*, in this paper. The key idea of CARA is that the transmitter station combines adaptively the Request-to-Send/Clear-to-Send (RTS/CTS) exchange with the Clear Channel Assessment (CCA) functionality to differentiate frame collisions from frame transmission failures caused by channel errors. Therefore, compared with other open-loop rate-adaptation schemes, CARA is more likely to make the correct rate adaptation decisions. Moreover, CARA does not require any change to the current 802.11 standard since both RTS/CTS mechanism and CCA functionality are the core parts of the 802.11 protocol, and hence are ready to use. This facilitates its deployment with existing 802.11 devices.

The rest of the paper is organized as follows. Related work is presented in Section II. Relevant issues and scheme details of CARA are described in Sections III and IV, respectively. Section V presents the simulation results, and finally, the paper concludes with the future work in Section VI.

1-4244-0222-0/06/$20.00 (c)2006 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the Proceedings IEEE Infocom.

II. Related Work

There have been remarkable studies on rate adaptation in the 802.11 WLANs. A transmitter station can change its transmission rate with or without feedback from the receiver, where the feedback information could be either Signal-to-Interference/Noise Ratio (SINR) or the desired transmission rate determined by the receiver. Depending on whether to use the feedback from the receiver, rate adaptation schemes can be classified into two categories: *closed-loop* and *open-loop* approaches.

In closed-loop approaches [13], [18], after the receiver specifies its desired transmission rate and feeds back to the transmitter as part of a modified RTS/CTS exchange, the transmitter adapts its transmission rate accordingly. Since the rate adaptation is dictated by the receiver, this approach does not suffer from frame collisions. However, in order to support such a feedback loop, the CTS (and possibly RTS) frame format should be modified to convey the extra information, which does not conform to the 802.11 standard. Moreover, using the RTS/CTS exchange itself is a costly solution, which wastes the precious wireless bandwidth when hidden stations do not exist. It should be noted that the RTS/CTS exchange is rarely used in the practical infrastructure-based WLANs due to this fact whereas it is highly desirable to use it all the time in multi-hop ad-hoc networks due to the existence of hidden stations.

With open-loop approaches, a transmitter station makes the rate adaptation decision solely based on its local Acknowledgment (Ack) information. In the 802.11 standard, an Ack frame is transmitted by the receiver upon successful reception of a data frame. It is only after receiving an Ack frame correctly that the transmitter assumes a successful delivery of the corresponding data frame. On the other hand, if an Ack frame is received in error or no Ack frame is received at all, the transmitter assumes failure of the corresponding data frame transmission. Open-loop approaches do not require any interaction between the transmitter and the receiver, and hence, is standard-compliant in general.

Open-loop approaches can be further classified into two subcategories. The first subcategory decides the transmission rate based on local channel estimation, e.g., made during Ack frame receptions, assuming a symmetric wireless channel between the transmitter and the receiver [12], [19], [20]. The schemes in this subcategory often yield very good performance similar to that of closed-loop approaches, but usually require extra implementation efforts. In contrast, the second subcategory only makes use of the local Ack information when selecting the transmission rate [11], [15], [21], which is very simple to implement. That is also the main reason why the ARF algorithm, belonging to the second subcategory, is adopted by the most commercial 802.11 WLAN products.

It has been pointed out in [21] that there are two fundamental issues when designing a rate adaptation scheme, i.e., *when to increase* and *when to decrease* the transmission rate. The effectiveness of a rate adaptation scheme depends greatly on how fast it may respond to the wireless channel variation. The schemes presented in [11], [21] address the first issue, and enhance the original ARF by allowing a transmitter station to increase its rate in an adaptive manner over a time-varying wireless channel. In this paper, we consider the other issue, i.e., *when to decrease the transmission rate*. To our best knowledge, most (if not all) existing open-loop rate adaptation schemes malfunction severely when there are many contending stations in the network causing a lot of frame collisions.[1] This is due to the fact that these schemes do not differentiate frame collisions from frame transmission failures caused by channel errors, and hence, may decrease the transmission rate over-aggressively.

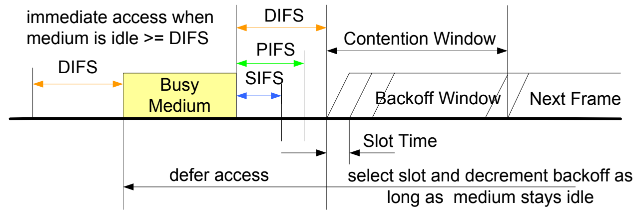


Fig. 1. Basic channel access mechanism of IEEE 802.11 DCF

III. Preliminaries

A. IEEE 802.11 CSMA/CA

IEEE 802.11 standard specifies two different MAC schemes: the mandatory DCF and the optional PCF [1]. Today, most 802.11 WLAN devices implement only the DCF due to its simplicity. The DCF is based on CSMA/CA as illustrated in Fig. 1. When a station is ready to transmit a frame, it checks the status of the channel. If the channel is busy, it waits until the end of the on-going transmission. This part makes the DCF a carrier sense multiple access (CSMA) protocol. Then, when the channel becomes idle, instead of transmitting immediately, the station selects a random backoff interval in order to reduce the collision probability. This part makes the DCF a collision avoidance (CA) protocol.

Even with random backoff, a transmitted frame may still collide with other frames when two or more stations finish the backoff simultaneously. Such frame collisions cannot be completely eliminated due to the contention nature of the DCF, and the problem becomes worse as the number of contending stations increases. Besides collision, a frame transmission failure may be caused by channel errors as well.

The mechanism for the PHY to determine whether the channel is busy or idle is called *Clear Channel Assessment (CCA)*. The CCA busy is declared by the underlying PHY when an energy level measured at the antenna front-end is above a threshold and/or a known carrier is detected. Obviously, the CCA mechanism is a core part of the CSMA/CA protocol.

[1] This is true even for the scheme proposed in [12] as one can easily verify it via simulations even if such a fact was never reported rigorously in the literature.

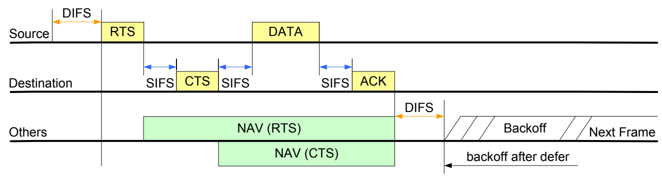


Fig. 2. RTS/CTS exchange mechanism of IEEE 802.11

In this paper, we propose that the transmitter station uses the CCA mechanism to detect the collision of its own frame transmission with other frame transmission(s).

B. RTS/CTS Exchange in IEEE 802.11

When hidden stations exist in the network, the performance of the basic CSMA/CA can be severely degraded. The unprotected time interval, however, can be shortened to the RTS transmission time, by preceding the data frame transmission with the exchange of two short control frames, i.e., RTS and CTS frames, and hence the hidden station problem can be ameliorated. This is known as the original objective of the RTS/CTS exchange. The RTS/CTS exchange is illustrated in Fig. 2, in which the wireless channel is reserved for the transmitter station after a successful exchange of RTS/CTS frames. According to the 802.11 standard [1], the decision to use the RTS frame transmission is made solely at the transmitter side. That is, the RTS frame is used when the size of the pending data frame is equal to or larger than the *RTS threshold* value. However, in most of the typical 802.11 devices operating in infrastructure-based WLANs with access points (APs), the RTS threshold is set to the largest value, i.e., 2347 octets, which basically disables the usage of RTS/CTS exchange. Accordingly, the RTS and CTS frames are rarely observed in the real WLANs.

It is also known that the RTS/CTS exchange is useful in heavily-contending WLAN environments, where many transmissions might fail due to collisions, and the advantage could be amplified with relatively large data frames [7]. The author of [7] also presents that the additional amount of time wasted due to collisions is negligible for the RTS/CTS mechanism, regardless of the number of contending stations and collision probability.

Although it is originally defined in [1] that the transmission of an RTS frame should be triggered based on the RTS threshold, using the RTS/CTS frames for other possible purposes, not restricted to the original definition, can be found in supplementary standards. According to the emerging IEEE 802.11e standard [5], the RTS/CTS exchange can be used independently of the RTS threshold. For example, the RTS frame can be transmitted to reserve a time interval, called transmission opportunity (TXOP), for the consecutive transmissions of multiple data frames. Meanwhile, an 802.11g transmitter can initiate an RTS/CTS exchange or simply transmit a CTS frame with the receiver address equal to itself in order to reserve the channel for the transmission of non-basic rate, i.e., Orthogonal Frequency Division Multiplexing (OFDM)-modulated, frames to address the co-existence problem between the 802.11g and legacy 802.11b devices [9].

In this paper, we propose an adaptive usage of the RTS/CTS exchange as a means to probe the channel status in order to differentiate frame collisions from transmission failure caused by channel errors. Even though the use of the RTS/CTS exchange for channel probing is not compliant to the IEEE 802.11 standard in a strict sense, it is hard to judge that our usage violates the 802.11 standard since the original restriction of the RTS/CTS exchange usage fades already under the influence of other usage cases specified in supplementary standards, e.g., the 802.11e and 802.11g, as addressed above. Moreover, since the support of the RTS/CTS exchange along with the implementation of these frames is a mandatory part of the 802.11 standard, our approach is readily implementable in the existing 802.11 devices, which is a key advantage of our approach.

C. ARF in IEEE 802.11

In the 802.11 market, the most widely implemented rate adaptation scheme is ARF, which was originally developed for Lucent Technologies' WaveLAN-II WLAN devices [15]. It alternates the transmission rates by keeping track of a timing function as well as missing Ack frames. If two consecutive Acks are not received correctly by the sender, the second retry of the data frame and the subsequent transmissions are done at a lower transmission rate and a timer is started. When either the timer expires or the number of successfully-received Acks reaches 10, the transmission rate is raised to the next higher transmission rate and the timer is cancelled. However, if an Ack is not received for the very next data frame, the transmission rate is lowered again and the timer is restarted.

Apparently, ARF has a purely heuristic and conservative nature, and hence it cannot react quickly when the wireless channel condition fluctuates. In other words, the transmitter station may attempt increasing its transmission rate to probe the wireless channel condition upon consecutive successful Ack receptions and decreasing its rate upon consecutive (re)transmission failures without any consideration of the actual cause of the transmission failures, i.e., channel errors or frame collisions. However, thanks to its simplicity, ARF is still widely employed in commercial 802.11 WLAN devices, and many proposed open-loop rate adaptation schemes, e.g., [11], [21], are rooted in ARF.

IV. Collision-Aware Rate Adaptation (CARA)

In this section, we present the details of our collision-aware rate adaptation scheme, called *CARA*. One salient feature of CARA is that it is able to differentiate collisions from channel errors at the transmitter side without any help/feedback from the receiver station.

Many rate adaptation schemes have been proposed in the 802.11 WLANs in order to fully exploit the multiple transmission rates, and ARF is one of the most widely-adopted rate adaptation schemes. Unfortunately, most (if not all) open-loop rate adaptation schemes including ARF do not work

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the Proceedings IEEE Infocom.

properly when multiple users contend for the shared wireless medium, since they are unable to identify the reason, i.e., frame collisions or channel errors, why an expected Ack frame is not received after a corresponding data frame transmission.

A. Identifying Collision via RTS Probing

CARA specifies two methods to differentiate collisions from channel errors, and the mandatory one is called *RTS Probing*. We assume that the transmission error probability of an RTS frame is negligible, because of its small size and robust transmission rate, and hence all the RTS transmission failures are due to collisions. On the other hand, we know that a data transmission failure following a successful RTS/CTS exchange must be due to channel errors, because the successful RTS/CTS exchange has already reserved the wireless channel and guarantees no collision to the subsequent data transmission. Therefore, if we exchange RTS/CTS frames before each data transmission and then apply the ARF scheme, there will be no data frame collisions, and hence no misinterpretation of a data frame collision as a channel-error-caused data frame transmission failure. As a result, unnecessary rate decrements are completely avoided. One side effect of this approach is the added RTS/CTS overhead, which wastes the precious wireless bandwidth. In fact, the RTS/CTS option is disabled in most 802.11 products currently available in the market. Based on the above observation, instead of mandating an RTS/CTS exchange before each data frame transmission, we propose *RTS Probing* which enables RTS/CTS exchange only when a data frame transmission fails.

1) State Transition Diagram: The detailed procedure of RTS Probing is best explained with a transmitter station's state transition diagram as shown in Fig. 3, where related symbols/parameters are listed in Table I. There are four states in the diagram:

- *Initial State*: the starting point of the procedure.
- *Wait for MPDU*: the station is in this state when there are new data frames coming from the upper layer or when the current frame transmission fails and retransmission is requested.
- *DATA Tx*: the station is in this state when it finishes a data transmission and awaits the corresponding Ack frame.
- *RTS Tx*: the station is in this state after it finishes an RTS transmission and awaits the corresponding CTS frame.

As shown in Fig. 3 and Table I, the consecutive failure count, n, is compared with two different thresholds, namely, the probe activation threshold (P_{th}) and the consecutive failure threshold (N_{th}), for different purposes. When n reaches P_{th}, the RTS/CTS frames will be exchanged before the next data retransmission attempt, while when n reaches N_{th}, the next data retransmission attempt will be conducted at a lower rate. With different values for P_{th} and N_{th}, the RTS Probing procedure works differently, and the default values for P_{th} and N_{th} are 1 and 2, respectively, in our scheme. Some example scenarios are explained as follows:

- $P_{th} = 0$: in this case, the RTS/CTS frames are exchanged before each data (re)transmission attempt. When an RTS/CTS exchange succeeds, the data frame is (re)transmitted. The data transmission rate falls back to the next lower level, if available, upon N_{th} data transmission failures.
- $P_{th} \geqslant 1, N_{th} = 1$: in this case, a data frame is transmitted without RTS/CTS support, and its rate falls back upon a single transmission failure. Note that RTS/CTS exchange is never activated whenever $P_{th} \geqslant N_{th}$.
- $P_{th} \geqslant 2, N_{th} = 2$: in this case, the rate falls back upon two consecutive transmission failures without RTS/CTS support. This is equivalent to ARF if $M_{th} = 10$.
- $P_{th} = 1, N_{th} = 2$ (the default values of CARA): in this case, a data frame is first transmitted without RTS/CTS support. If the transmission fails, the RTS/CTS exchange will be activated for the next retransmission attempt, and the transmission rate falls back if the retransmission fails again.

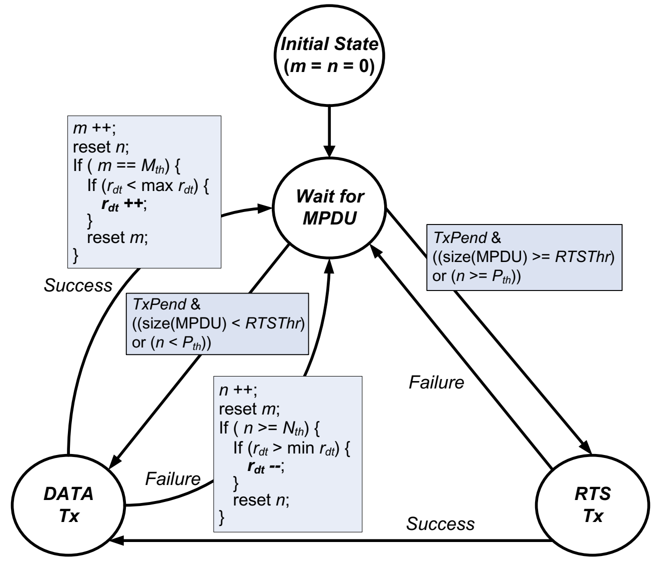


Fig. 3. State transition diagram of RTS Probing

TABLE I

LIST OF NOTATIONS USED IN THE RTS PROBING PROCEDURE

Notations	Comments
m	consecutive success count
n	consecutive failure count
M_{th}	consecutive success threshold
N_{th}	consecutive failure threshold
TxPend	status: a data frame is pending
R_{dt}	array of transmission rates 802.11a = {6, 12, 18, 24, 36, 48, 54 Mbps}* 802.11b = {1, 2, 5.5, 11 Mbps}
r_{dt}	transmission rate: an element of R_{dt}
$++$	increase transmission rate to the next higher one
$--$	decrease transmission rate to the next lower one
P_{th}	probe activation threshold
RTSThr	frame size-based RTS Threshold as defined in the standard

* The 9 Mbps rate is excluded as it is shown useless in [19].

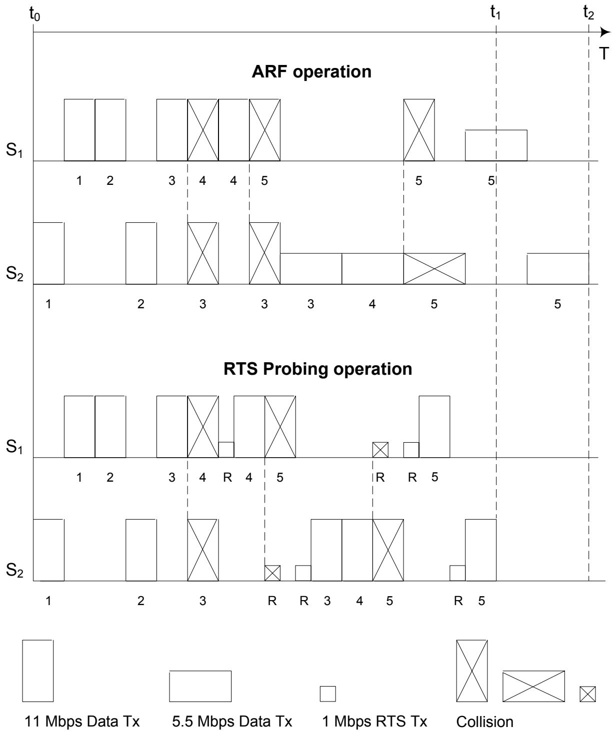


Fig. 4. Illustration of ARF and RTS-Probing timelines for a two-station network, when channel status is good enough to accommodate the highest transmission rate of the 802.11b PHY, i.e., 11 Mbps

It is interesting to see that, if the wireless channel condition suddenly becomes so bad that both RTS and Data transmissions fail, the transmitter station may be stuck in the *(Wait for MPDU)* $\mapsto$ *(RTS Tx)* $\mapsto$ *(Wait for MPDU)* loop forever. Fortunately, in this situation, since data frames are more vulnerable to channel errors, they would never be transmitted successfully. Therefore, there is no undesired side effect caused by the existence of such a loop. Once the wireless channel recovers from the bad state and after an RTS frame is delivered successfully, the data frame transmission attempts may resume.

Another threshold in the figure — the consecutive success threshold (M_{th}) — represents the number of consecutive successful frame transmissions that a transmitter station needs to observe before increasing its transmission rate. Since we focus on *when to decrease the transmission rate* in this work, we simply set M_{th} to be the same value as in ARF: $M_{th} = 10$. We will consider how to adapt this threshold in time-varying channel environments, e.g., as studied in [11], [21], in the future as a complement to the current version of CARA.

2) Examples: We use some simple examples to illustrate the RTS Probing procedure, and compare it with the ARF scheme in Fig. 4. Assume that two 802.11b stations S_1 and S_2 are contending for the shared wireless medium with the same data frame size, and the channel condition is good enough to accommodate the highest transmission rate of 11 Mbps. Successful transmissions are shown as blank rectangles in the figure, while the crossed rectangles represent frame collisions. All the inter-frame spaces, backoff durations, and CTS/Ack transmissions are omitted in the figure for simplicity.

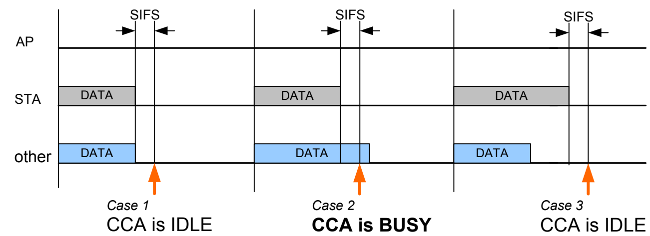


Fig. 5. Three possible cases of collision. In the second case, the collision can be detected via CCA detection.

It is clear that, with the ARF scheme, after four data transmission attempts, S_2's transmission rate falls back to 5.5 Mbps, while, with RTS Probing, S_2 is still able to preserve its high transmission rate at 11 Mbps. Moreover, we can see that, with RTS Probing, five frame transmissions are completed at an earlier time than that when ARF is used, i.e., $t_1 < t_2$, meaning that more frame transmissions may be accommodated with CARA with RTS Probing, and better channel utilization may be achieved.

B. Identifying Collision via CCA Detection

The second method to differentiate collisions from channel errors is called *CCA Detection*, which is optional in CARA and serves as a supplement to RTS Probing.

Before describing the details of CCA Detection, let's take a look into Fig. 5 that illustrates three possible collision scenarios, with the assumption of no hidden stations. In this figure, "STA" is the transmitter station of our interest while "other" represents another station which starts its frame transmission simultaneously with STA. Actually, this figure could also represent multiple frame collisions, and in such cases, "other" represents the station transmitting the frame with the longest transmission time among multiple colliding frames. Collisions can be classified into three cases as shown in Fig. 5: in the first case, the colliding frames have the same transmission duration, while the second and third cases illustrate different transmission durations.

In general, the CCA function in the 802.11 is used by a wireless station to assess the channel occupancy status at a given time. For example, in the case of CCA busy, the station freezes its backoff process. On the other hand, in the case of CCA idle, the backoff process may be resumed or the station may start transmitting if the backoff count reaches zero. Our CCA Detection method works as follows. At SIFS time after a wireless station finishes its data transmission, it starts assessing the wireless channel using CCA. Since the station expects an Ack reception at this time point, so if the wireless channel is assessed as busy while the expected Ack reception does not start, the station concludes that a collision has just happened to its data transmission, which corresponds to Case 2 in Fig. 5. In this case, the transmitter station would retransmit without increasing failure count (n) and lowering the transmission rate. Note, though, that CCA Detection will not help in Case 1 or Case 3. Therefore, in these cases, the RTS Probing procedure is launched after the CCA Detection procedure fails to detect any collision explicitly.

One should be noted that a station operating at 1 Mbps is likely to experience Case 3 more often than Case 1 or Case 2 for a given distribution of data frame sizes since its transmission duration is longer than others operating at higher transmission rates if the frame sizes are the same. Accordingly, CCA Detection is less likely to occur when a transmitter station operates at 1 Mbps. However, knowing that 1 Mbps is the lowest rate so that no further rate decrease is needed, this is a nice property for CCA Detection.

V. PERFORMANCE EVALUATION

In this section, we evaluate the effectiveness of CARA by using the ns-2 simulator [23] after enhancing the original 802.11 DCF module to support the 802.11b PHY and the time-varying wireless channel model.

A. Simulation Setup

We mainly simulate an infrastructure-based 802.11b system except for one case in which the ad-hoc mode is simulated. Each station transmits with 20 dBm power, and all the stations are static. We use the empirical BER (Bit Error Rate) vs. SNR (Signal-to-Noise Ratio) curves, provided by Intersil [22],[2] to estimate the FER (Frame Error Rate). The background noise level is set to -96 dBm. Besides, we use a log-distance path-loss model with the path-loss exponent of four [17] to simulate the indoor office environments.

Moreover, we also consider the multi-path fading effect with which the channel condition between the transmitter and receiver varies over time. We use the Ricean fading model [16] to simulate the time-varying wireless channel conditions. The Ricean distribution is given by:

$$p(r) = \frac{r}{\alpha^2} e^{\left(-\frac{r}{2\sigma^2}+K\right)} I_0(2Kr), \quad (1)$$

where K is the distribution parameter representing the line-of-sight component of the received signal, α^2 is the variance of the background noise, r is the received power, and $I_0(\cdot)$ is the modified Bessel function of the first kind and zero order [17]. The frequency of the channel condition change depends on the relative speed of the mobile station with respect to its surroundings. Note that the channel fluctuation can occur due to the moving environment even if the station itself does not move. We assume 2.5 m/s velocity (of the moving environment) for our simulations, and this corresponds to the Doppler spread of 20 Hz.

We evaluate the following testing schemes: (1) single-rate schemes (referred to as Rx) using fixed transmission rate x Mbps (x = 1, 2, 5.5, 11); (2) the ARF scheme (referred to as *ARF*); (3) the ARF scheme using the RTS/CTS exchange all the time (referred to as *RTS/CTS*); and (4) our proposed rate adaptation schemes, CARA-1 (with only the mandatory RTS Probing) and CARA-2 (with both the mandatory RTS Probing and the optional CCA Detection). The testing schemes are compared with each other in terms of the aggregate system throughput (in Mbps).

[2]The BER curves in [22] are measured in an AWGN (Additive White Gaussian Noise) environment without any fading channel.

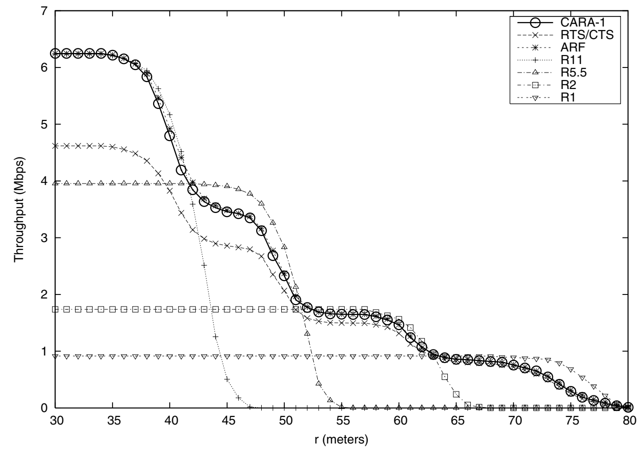


Fig. 6. Throughput comparison of our proposed rate adaptation scheme (CARA-1) against RTS/CTS, ARF, and single-rate schemes for one-to-one topology networks with various distance (r)

As addressed in Section IV, we set the consecutive success threshold (M_{th}) to 10, and the consecutive failure threshold (N_{th}) to 2, for both ARF and CARA. Moreover, we set the probe activation threshold (P_{th}) to 1 in CARA. The RTS/CTS frames are always transmitted at the lowest rate of 1 Mbps.

We conduct the simulations under various network topologies, data frame sizes, and channel models. Each station transmits in a greedy mode, i.e., its data queue is never empty, and all the data frames are transmitted without fragmentation. We use LLC/IP/UDP as the upper layer protocol suite, and the MAC-layer data payload length is 1500 octets unless specified otherwise.

B. Results for One-to-One Topology

We first compare the testing schemes in the simplest one-to-one topology, in which one station continuously transmits frames to the other station, i.e., the AP, with various distance r ($30 \leqslant r \leqslant 80$) meters between two stations. In this simulation, an AWGN wireless channel model is assumed. Simulation results are plotted in Fig. 6, where the thick solid line with circle points represents CARA-1. We omit the performance of CARA-2 since both CARA-1 and CARA-2 perform exactly the same in this environment. Note that the optional CCA detection of CARA-2 will not work at all when there is no frame collision.

In general, the throughput decreases for all testing schemes as the distance increases. R1 is the most conservative scheme of all. It transmits all the frames at the lowest 1 Mbps, and hence it results in the lowest throughput when r is small. At the same time, due to the strong error-correcting capability of

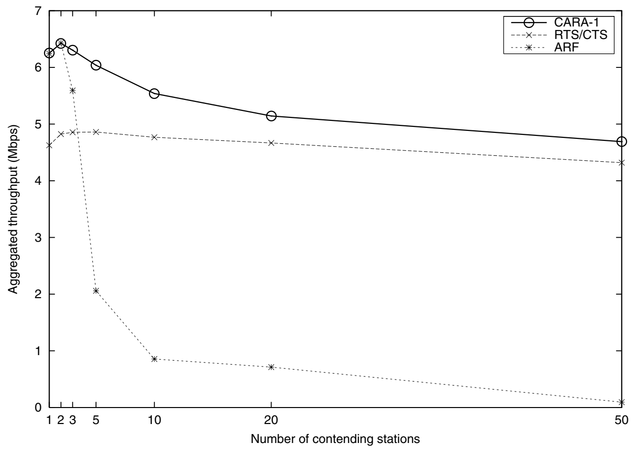


Fig. 7. Throughput comparison of our proposed rate adaptation scheme (CARA-1) against RTS/CTS and ARF for star-topology networks with various number of contending stations

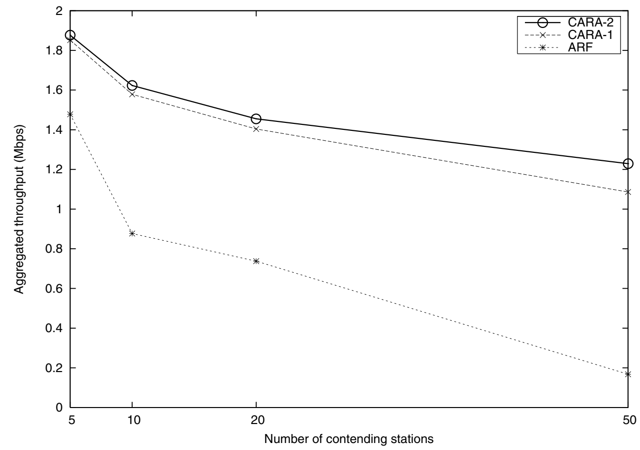


Fig. 8. Throughput comparison of our proposed rate adaptation schemes (CARA-1 and CARA-2) against ARF for line-topology networks with various number of contending stations with randomly chosen data frame sizes and stations' positions

1 Mbps mode, even when the transmitter is far away from the receiver, they can still communicate successfully. On the other hand, R11 is the most aggressive scheme, which transmits all the frames at the highest 11 Mbps. R11 allows the transmitter station to make better use of the available bandwidth when r is small. However, due to the poor error-correcting capability of 11 Mbps mode, the throughput degrades drastically as r increases. In fact, when $r > 47$, all the transmission attempts fail, and the throughput drops to zero. Other single-rate schemes can be viewed as compromises between R1 and R11.

Note that in this topology without contention, one of the single-rate scheme is supposed to perform the best for a given distance. We observe that CARA-1 basically performs close to the best single-rate scheme for the entire range of the distance in consideration. That is, its throughput curve follows the outer envelope of those of the single-rate schemes very well. Moreover, thanks to its adaptive activation of the RTS/CTS frame exchange, CARA-1 achieves comparable throughput with ARF. In comparison, the RTS/CTS scheme yields significantly lower throughput than both ARF and CARA-1 due to added overhead of RTS/CTS frame exchanges before each data transmission attempt.

C. Results for Star Topologies with Varying Number of Contending Stations

We now consider the star-topology networks with varying number of contending stations in order to study the collision effect on the system performance. With this scenario, various number of contending stations are evenly spaced on a circle around the AP with the radius of 10 meters, and all the stations are static. Similar to the previous simulation, an AWGN wireless channel model is assumed. It should be noted that in this environment, i.e., the distance of 10 meters, the stations should be able to transmit the data frames at 11 Mbps all the time as we can easily conclude from Fig. 6.

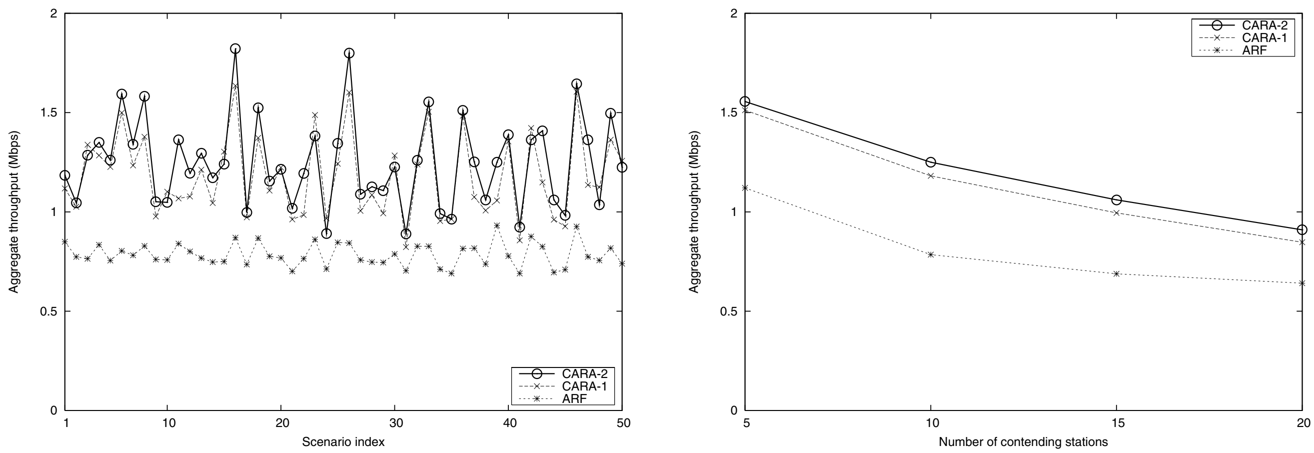
Simulation results are plotted in Fig. 7. Both RTS/CTS and ARF yield lower throughput than CARA-1 in all simulated scenarios. In particular, with ARF, the aggregate system throughput is degraded severely even with a small number of contending stations in the network. For example, when the number of contending stations increases from 2 to 5 and 10, the aggregate throughput with ARF drops from over 6 Mbps to about 2 Mbps and under 1 Mbps, respectively. RTS/CTS does not work as poorly as ARF, i.e., it continues to work well even with many contending stations. However, it performs worse than CARA-1 since it wastes the wireless bandwidth by exchanging the RTS/CTS frames before each data frame transmission attempt.

There are two main reasons for ARF's ill behavior. First, since ARF cannot differentiate collisions from channel errors, a wireless station may decrease its frame transmission rate over-aggressively, and then operate with a lower transmission rate than the actual achievable higher rate. Second, since each contending station conducts its rate adaptation independently, they may end up transmitting data at different rates. Such transmission-rate diversity causes the following *performance anomaly* that was first discovered experimentally by the authors of [14]: since the 802.11 DCF is designed to offer equal transmission opportunities (or long-term equal medium access probabilities) to all contending stations, the throughput of a high-rate station is always bounded below the lowest transmission rate in the network.

In comparison, CARA-1 achieves significantly higher, i.e., about 11.5 times (on average), aggregated system throughput than ARF in this simulation setup.

D. Results for Line Topologies with Random Data Frame Sizes and Random Station Positions

We now compare CARA-1 and CARA-2 against ARF in an environment with varying number of contending stations.



(a) 50 different scenarios when 10 stations contend

(b) Averaged results with various number of contending stations

Fig. 9. Throughput comparison of our proposed rate adaptation schemes (CARA-1 and CARA-2) against ARF for random-topology networks (with randomly chosen data frame sizes)

Recall that both RTS Probing and CCA Detection are implemented in CARA-2. CCA Detection only helps when the data transmission durations from contending stations are different and when there are no hidden stations in the network. In order to show this effect, we allow each contending station to select its data frame size randomly for each frame. Besides, we simulate a line topology with the AP sitting at one end, and all contending stations are randomly placed along the line. The maximum distance between a wireless station and the AP is set to 70 meters in order to guarantee no hidden stations. This is because, according to Fig. 6, two stations are able to communicate robustly with each other at the most reliable rate of 1 Mbps when they are less than 70 meters apart. Again, we simulate an AWGN wireless channel model.

Simulation results are plotted in Fig. 8, where each point is averaged over 50 simulation runs. As shown in the figure, when the number of stations increases, the performance gap between CARA-2 and CARA-1 gets larger because, with more contending stations in the network, it is more likely that the colliding data frames have different transmission durations. Consequently, CCA Detection may be more helpful in differentiating collisions from channel errors under those circumstances. On the other hand, ARF yields significantly worse throughput than CARA-1 and CARA-2 in all simulated scenarios.

E. Results for Random Topologies with Time-Varying Wireless Channel

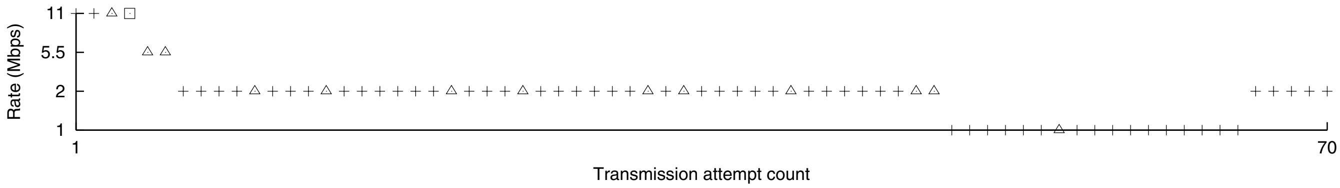
We also evaluate and compare the performances of the testing schemes in randomly-generated network topologies: all the transmitter stations and their (different) respective receivers are randomly placed within a circle around the AP with the radius of 40 meters. Each station selects a random data frame size. In this simulation, we assume a Ricean fading channel [16] with Ricean K factor of 3 dB to describe the indoor fading channel environment [8]. We simulate 50 different scenarios with 10 contending stations in the network, and the results are plotted in Fig. 9(a). Furthermore, we simulate with various number of contending stations, and Fig. 9(b) shows the simulation results with each point averaged over 50 random topologies.

We have two observations from the figures. First, both CARA-1 and CARA-2 are significantly better than ARF, in terms of aggregate system throughput, in each simulated random topology, while CARA-2 outperforms CARA-1 (on average) regardless of the number of contending stations. Second, different from what we have observed in Section V-D with an AWGN channel, CARA-2 does not always outperform CARA-1 with a Ricean fading channel. As shown in Fig. 9(a), CARA-1 yields higher throughput than CARA-2 in 8 out of 50 simulated scenarios. This is surprising at the first sight but rather reasonable for the following reason. Consider the situation when the wireless channel condition suddenly turns bad after a successful CCA detection by CARA-2. As a result, the transmitter station may fail to retransmit its data frame. However, the consecutive failure count n is not increased due to the successful CCA detection. As a result, the future rate adaptation, i.e., rate decrement, will be delayed, which may cause more data frame transmission failures, and result in lower aggregate throughput.

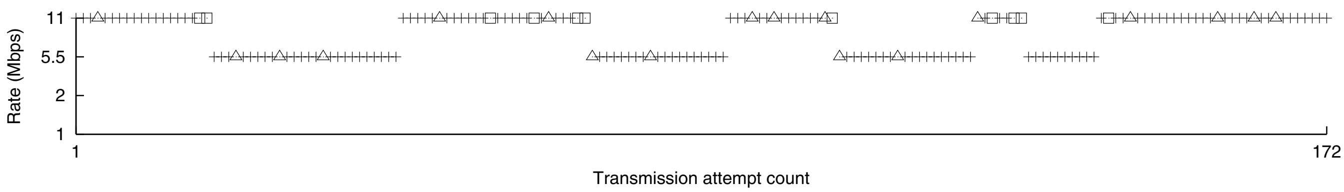
F. Transmission Rate Adaptation over Time

We now consider the behavior of three testing schemes over time. The simulation setup is similar to that in Section V-C. That is, we simulate a star topology, where five stations are evenly spaced on a circle around the AP with the radius of 40 meters, and the AWGN wireless channel is assumed. As we observe from Fig. 6, each transmitter is expected to alternate between 11 Mbps and 5.5 Mbps at this distance .

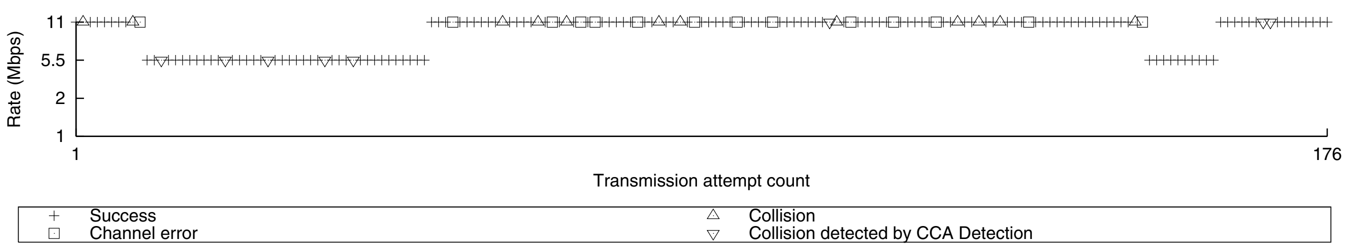
The transmission rate selections by one of five stations, for a given simulation time interval of 1.6 seconds, under



(a) ARF



(b) CARA-1



(c) CARA-2

Fig. 10. Adaptability comparison of ARF and our proposed rate adaptation schemes (CARA-1 and CARA-2) when 5 stations are contending

ARF, CARA-1, and CARA-2 are shown in Figs. 10(a), 10(b), and 10(c), respectively. The x-axis represents the data frame transmission attempt count, which excludes the RTS transmission attempts. Note that the maximum frame transmission count for each scheme is different for the same simulation time of 1.6 seconds. This means that different numbers of frame transmission attempts were made for different schemes, and this should be due mainly to the employed transmission rates. In this figure, the following symbols are used to represent a data frame transmission result:

- *Cross point*: a successful data frame transmission
- *Square point*: a data transmission failure caused by channel error
- *Triangle point*: a data frame collision
- *Inverse triangle point*: a data frame collision identified by CCA Detection

It should be noted that after each triangular point of CARA-1 and CARA-2, there is at least one RTS transmission attempt due to RTS Probing while such RTS transmission attempt is not illustrated in the figures.

We have three observations. First, ARF could not differentiate collisions from channel errors, and hence, decreases the transmission rate over-aggressively. For example, we observe that ARF decreases the rate from 5.5 Mbps to 2 Mbps and then from 2 Mbps to 1 Mbps after two consecutive frame collisions, respectively. It operates at the low rates of 1 or 2 Mbps for most of the time. Second, CARA-1 and CARA-2 are better at adapting to the wireless channel condition even when frame collisions are present. We can see that both CARA-1 and CARA-2 operate at the high rates of 11 or 5.5 Mbps. Finally, CARA-2 makes more frame transmission attempts than CARA-1 during the same time interval of 1.6 seconds, and this generally implies more successful data frame transmissions, thus resulting in higher throughput. Table II compares the testing schemes in terms of the numbers of frame transmission attempts, transmission successes, and the corresponding throughput for the entire simulation run of 30 seconds. We here reconfirm that CARA-2 outperforms CARA-1 thanks to its capability of CCA detection.

G. Summary

Based on the observations from the simulation results, we summarize the effectiveness of CARA as follows:

- RTS Probing is very effective in differentiating collisions

TABLE II
COMPARISON OF THREE TESTING SCHEMES FOR THE 30-SECOND SIMULATION RUN

	ARF	CARA-1	CARA-2
# of tx attempts	1344	3092	3246
# of tx successes	1094	2518	2643
Throughput (Mbps)	1.58	3.37	3.49

from channel errors. For this reason, CARA outperforms ARF significantly in terms of aggregate throughput.

- With the additional help from CCA Detection, CARA-2 yields even higher aggregate throughput than CARA-1 when the data transmission durations are different among contending stations.
- Having observed the ill behavior of the ARF scheme, which happens to be the most widely-deployed rate adaptation scheme in the commercial 802.11 devices, we conclude that *it is critical to have a well-designed link adaptation scheme with collision-awareness feature (e.g., CARA) to replace ARF such that the multiple transmission rates of an 802.11 device may be fully exploited and the throughput performance may be improved.*

VI. CONCLUSION AND FUTURE WORK

In this paper, we propose a novel collision-aware rate adaptation scheme, called *CARA*, for IEEE 802.11 WLANs. The key idea of CARA is that the transmitter station combines adaptively the RTS/CTS exchange with the CCA functionality to differentiate frame collisions from frame transmission failures caused by channel errors. Therefore, compared with ARF, the most well-known and widely-deployed rate adaptation scheme in the commercial 802.11 WLAN devices, CARA is more likely to make the correct rate adaptation decisions. Moreover, CARA does not require any change to the current 802.11 standard, thus facilitating its deployment with existing 802.11 devices.

The performance of CARA is evaluated via in-depth simulations over various scenarios in terms of network topology, data frame size, and wireless channel model. It is demonstrated that CARA significantly outperforms ARF in all the simulated multiple contending station environments, whereas the performance enhancement becomes more and more evident as the number of contending stations increases.

In the future, we plan to conduct further research to enhance CARA as follows:

1) The current version of CARA mainly considers when to decrease the transmission rate. However, when to increase the rate, represented by the consecutive success threshold (M_{th}), is also very important, and it critically affects the performance of a rate adaptation algorithm, especially, in time-varying channel environments. We plan to combine CARA with a recently-proposed enhanced ARF [21], which adaptively increases the transmission rate, to make the rate adaptation scheme more complete.
2) We also plan to work on the optimization/adaptation of other operational parameters including the probe activation threshold (P_{th}) and the consecutive failure threshold (N_{th}). While we use the default values of 1 and 2 for these thresholds, respectively, in this work, we expect that the performance can be further enhanced by optimizing and/or adapting these values. This will be done along with the evaluation of CARA over other higher-speed PHYs such as the 802.11a and the 802.11g, which provide more diversified transmission rate sets.
3) Note that we intentionally avoided the hidden-station environments in our performance evaluation since using the RTS/CTS exchange all the time is desirable in such environments. However, we expect that CARA will work relatively well even in such environments thanks to its adaptive usage of the RTS/CTS exchange. We plan to further enhance CARA by enabling the RTS/CTS exchange per detection of hidden stations, where such detection will become reality with the emerging 802.11k for radio resource management [6].
4) Finally, as a proof-of-concept, we plan to make a prototype by implementing CARA into the real WLAN devices. Thanks to the availability of an open source platform, i.e., MADWIFI [24], which allows us to modify lots of MAC operational configurations, we expect it viable.

APPENDIX

The abbreviations and acronyms used in the paper are summarized in alphabetic order in Table III.

REFERENCES

[1] IEEE 802.11, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, IEEE Std. 802.11-1999, Aug. 1999.

[2] IEEE 802.11b, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-speed Physical Layer Extension in the 2.4 GHz Band*, Supplement to IEEE 802.11 Standard, Sep. 1999.

[3] IEEE 802.11a, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-speed Physical Layer in the 5 GHz Band*, Supplement to IEEE 802.11 Standard, Sep. 1999.

[4] IEEE 802.11g, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Further Higher Data Rate Extension in the 2.4 GHz Band*, Supplement to IEEE 802.11 Standard, Jun. 2003.

[5] IEEE 802.11e/D13.0, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical layer (PHY) specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS)*, Draft Supplement to IEEE 802.11 Standard, Draft 13.0, Jan. 2005.

[6] IEEE 802.11k/D2.0, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical layer (PHY) specifications: Radio Resource Measurement*, Draft Supplement to IEEE 802.11 Standard, Draft 2.0, Feb. 2005.

[7] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal of Selected Area in Communications*, vol. 18, no. 3, March 2000.

[8] M. Carroll, T. A. Wysocki, "Fading Characteristics for Indoor Wireless Channels at 5GHz Unlicensed Bands," in *Proc. SympoTIC'03*, pp 102-105, Bratislava, Slovakia, Oct. 2003.

[9] Sunghyun Choi and Javier del Prado, "802.11g CP: A Solution for IEEE 802.11g and 802.11b Inter-Working," in Proc. IEEE VTC'03-Spring, Jeju, Korea, April 2003.

TABLE III

LIST OF ABBREVIATIONS AND ACRONYMS

Ack	Acknowledgement
AP	Access Point
ARF	Automatic Rate Fallback
BER	Bit Error Rate
CARA	Collision-Aware Rate Adaptation
CCA	Clear Channel Assessment
CSMA/CA	Carrier-Sense Multiple Access with Collision Avoidance
CTS	Clear-to-Send
DCF	Distributed Coordination Function
DIFS	Distributed Inter-Frame Space
FER	Frame Error Rate
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
LLC	Logical Link Control
MAC	Medium Access Control
MPDU	MAC Protocol Data Unit
MSDU	MAC Service Data Unit
NAV	Network Allocation Vector
OAR	Opportunistic Auto Rate
OFDM	Orthogonal Frequency Division Multiplexing
PCF	Point Coordination Function
PIFS	PCF Inter-Frame Space
PHY	Physical Layer
RBAR	Receiver-Based Auto Rate
RTS	Request-to-Send
SIFS	Short Inter-Frame Space
SINR	Signal-to-Interference/Noise Ratio
SNR	Signal-to-Noise Ratio
TXOP	Transmission Opportunity
UDP	User Datagram Protocol
WLAN	Wireless Local-Area Network

[10] Sunwoong Choi, Kihong Park, and Chong-kwon Kim, “On the Performance Characteristics of WLANs: Revisited,” in *Proc. ACM SIGMETRICS’05*, Jun. 2005.

[11] P. Chevillat, J. Jelitto, A. Noll Barreto, and H. L. Truong, “A Dynamic Link Adaptation Algorithm for IEEE 802.11a Wireless LANs,” in *Proc. IEEE ICC’03*, Anchorage, AK, May 2003, pp. 1141-1145.

[12] Javier del Prado Pavon and Sunghyun Choi, “Link Adaptation Strategy for IEEE 802.11 WLAN via Received Signal Strength Measurement,” in *Proc. IEEE ICC’03*, Anchorage, AK, May 2003.

[13] Gavin Holland, Nitin Vaidya, and Paramvir Bahl, “A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks,” in *Proc. ACM MobiCom’01*, Rome, Italy, Jul. 2001.

[14] M. Heusse, F. Rousseu, G. Berger-Sabbatel, and A. Duda, “Performance Anomaly of 802.11b,” in *Proc. IEEE INFOCOM’03*, vol. 2, pp. 836-843, Mar. 2003.

[15] Ad Kamerman and Leo Monteban, “WaveLAN-II: a high-performance Wireless LAN for the Unlicensed Band,” *Bell Labs Technical Journal*, vol.2, no.3, pp.118-133, Aug. 1997.

[16] Ratish J. Punnoose, Pavel V. Nikitin, and Daniel D. Stancil, “Efficient Simulation of Ricean Fading within a Packet Simulator,” in *Proc. IEEE VTC’00-Fall*, Sept. 2000.

[17] T. S. Rappaport, *Wireless Communications: Principle and Practice*, Englewood Cliffs, NJ: Prentice-Hall, 1996

[18] B. Sadeghi, V. Kanodia, A. Sabharwal and E. Knightly, “Opportunistic Media Access for Multirate Ad Hoc Networks,” in *Proc. ACM MobiCom’02*, Atlanta, Georgia, Jul. 2002.

[19] Daji Qiao and Sunghyun Choi, “Goodput Enhancement of IEEE 802.11a Wireless LAN via Link Adaptation,” in *Proc. IEEE ICC’01*, Jun. 2001, pp. 1995-2000.

[20] Daji Qiao, Sunghyun Choi, and Kang G. Shin, “Goodput Analysis and Link Adaptation for IEEE 802.11a Wireless LANs,” *IEEE Trans. On Mobile Computing (TMC)*, vol. 1, no. 4, pp. 278-292, Oct.-Dec. 2002.

[21] Daji Qiao and Sunghyun Choi, “Fast-Responsive Link Adaptation for IEEE 802.11 WLANs,” in *Proc. IEEE ICC’05*, Seoul, Korea, May 2005.

[22] Intersil, “HFA3861B; Direct Sequence Spread Spectrum Baseband Processor,” January 2000.

[23] “The Network Simulator – ns-2,” http://www.isi.edu/nsnam/ns/, online link.

[24] MADWIFI, http://sourceforge.net/projects/madwifi/, online link.