Enhancing Voice over WLAN via Rate Adaptation and Retry Scheduling

Hyewon Lee, *Student Member, IEEE* , Seongho Byeon, *Student Member, IEEE* , Byoungjin Kim, Kwang Bok Lee, *Fellow, IEEE* , and Sunghyun Choi, *Fellow, IEEE*

Abstract—Today, voice over IP (VoIP) service is emerging as a popular and important application in wireless local area networks (WLANs). While rate adaptation (or link adaptation) has been identified as a key factor determining the performance of WLANs, we have observed that most (if not all) rate adaptation algorithms have been developed to improve the throughput of data traffic, not the quality of service (QoS) of VoIP traffic. Accordingly, in this paper, we investigate the characteristics of VoIP traffic and the limitations of state-of-the-art rate adaptation algorithms, and then enhance the QoS of voice over WLAN (VoWLAN) by ameliorating the existing rate adaptation algorithms. Specifically, we design *fast decrease* to control the transmission rate of retransmissions, and *retry scheduling* to avoid the deep fading of the wireless channel as well as hidden terminal interference. We comparatively evaluate the QoS of the revised rate adaptation algorithms via ns-3 simulations and MadWiFi implementations in various communication environments, and demonstrate that the proposed schemes improve the R-score performance by up to 80 percent depending on the network scenarios.

Index Terms—IEEE 802.11, WLAN, voice over IP (VoIP), quality of service (QoS), rate adaptation

✦

1 INTRODUCTION

T ODAY'S numerous mobile devices such as laptops, smartphones, and mobile tablet personal computers have been employing wireless local area network (WLAN) technology as their wireless access interfaces. The IEEE 802.11 standard, often referred to as WiFi [2], has been evolving accordingly in order to comply with the increased needs of users, and various types of services over WLAN are being provided today. One of the most popular and important services is voice over WLAN (VoWLAN), and numerous research has studied provisioning quality of service (QoS) for VoWLAN [3], [4], [5], [6], [7].

Currently, IEEE 802.11a/b/g/n physical (PHY) layers support a number of PHY transmission rates (or PHY rates) with different throughput and error rate performances [8], [9]. It is generally known that low transmission rates are more robust than high transmission rates, while high transmission rates can provide WLAN with higher throughput. Hence, *rate adaptation* (or *link adaptation*), which adaptively selects a PHY transmission rate with a specific objective in a given communication environment, has been an important research topic in WLANs [10], [11], [12], [13], [14], [15], [16]. The objective of rate adaptation actually can vary with communication environments, e.g., traffic type to be served, number of users, metric for network performance, etc. It should be noted that, however, most of up-to-date researches

on rate adaptation focus on maximizing the throughput of WLAN, not on guaranteeing the QoS of VoWLAN.

IEEE 802.11 medium access control (MAC) standard defines *retransmissions* of MAC frames in order to provide a better frame delivery probability against channel errors and/or frame collisions. Upon a frame transmission failure, which is recognized by the absence of a corresponding acknowledgement (ACK) frame response, MAC frames can be retransmitted up to the retry limit. However, continual retransmissions could be detrimental in some cases, as the QoS of voice traffic is known to be very sensitive to frame losses. For example, let us assume that a frame transmission fails due to the deep fading of the wireless channel and/or severe hidden terminal interference, where both of these events can frequently happen in practical WLAN environments. If a WLAN station continues to retransmit the failed frame within a short time interval in which the channel status remains poor, then the frame transmissions will continue to fail and the frame will be dropped eventually after a finite number of retransmissions. Thus, considering that the voice traffic is sensitive to frame losses, continual frame losses at the MAC layer can severely degrade the QoS of VoWLAN.

In this paper, we investigate state-of-the-art rate adaptation algorithms and discuss their QoS provisioning capabilities for VoWLAN. We also propose simple yet effective modifications to the existing rate adaptation algorithms to improve the QoS of VoWLAN. We further enhance the QoS of VoWLAN by scheduling the frame retransmissions, so that WLAN stations can cope with the deep fading and hidden terminal interference and provide a better delivery ratio of VoWLAN frames.

The rest of the paper is organized as follows. We first introduce the VoWLAN and investigate the existing rate adaptation algorithms of the WLAN in Section 2. Section 3 proposes two novel features for VoWLAN, namely, *fast decrease (FD)* and *retry scheduling (RS)*, to improve the QoS.

- H. Lee, S. Byeon, K.B. Lee, and S. Choi are with the Department of Electrical & Computer Engineering and INMC, Seoul National University, Seoul, Korea.
  E-mail: {hwlee, shbyeon}@mwnl.snu.ac.kr, {klee, schoi}@snu.ac.kr.
- B. Kim is with the Institute of Network Technology, Core Network Lab, SK Telecom, Korea. E-mail: byoungjin.kim@sk.com.
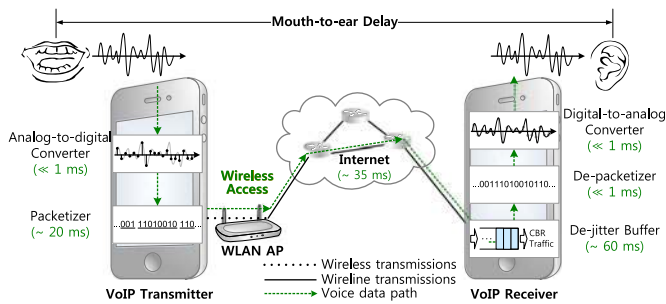
Fig. 1. An illustration of the mouth-to-ear delay. Delay values are derived assuming G.711/722 and AMR-WB voice codec with 20 ms inter-packet interval and 60 ms de-jitter buffer delay in a general Internet environment with a WLAN first-hop connection. The processing delays are assumed to be negligible.

In Sections 4 and 5, we comparatively evaluate the QoS of the proposed schemes in various communication environments via network simulator (ns-3) and MadWiFi implementation results, respectively [17], [18]. Finally, Section 6 discusses the effectiveness of the proposed schemes on enhancing the QoS of VoWLAN as well as further extensions, and then concludes the paper.

2 REVISIT OF VOICE OVER WLANs

As the VoWLAN service is getting popular, there has been intensive research on the VoWLAN, e.g., the call admission control of VoIP sessions [3], [4], [5], and the capacity enhancement of VoWLANs [6], [7]. In this paper, we specifically focus on the rate adaptation of the VoWLAN to enhance the QoS.

We first study the characteristics of VoIP traffic as well as its performance metric, *R-score*. Then we investigate the existing rate adaptation algorithms for WLANs and discuss their operations and limitations. We also explain the retransmission policy of the IEEE 802.11 MAC and how this affects the QoS of VoWLAN.

2.1 Characteristics of VoIP Traffic

Unlike the typical objective of non-real-time data traffic, i.e., maximizing its throughput, real-time VoIP traffic requires guaranteed QoS. Aiming at provisioning high QoS VoWLAN service, we first study the characteristics of VoIP traffic, and then discuss the means to evaluate its QoS.

2.1.1 Voice Codec and Mouth-to-Ear Delay

For VoIP, an analog voice signal is sampled and encoded using a voice codec, e.g., ITU-T G.711/722/723.1/729a [19], [20], [21], [22] and adaptive multi rate wide band (AMR-WB, G.722.2) [23], into a digital bit stream. The encoded signal is then packetized periodically and then is delivered to the MAC layer as a frame payload. Without silence suppression, which detects silent durations of human speech and does not generate any voice packet during that interval, the voice traffic is basically constant bit rate (CBR) traffic, i.e., voice packets of a fixed size are generated and transmitted periodically. After packet deliveries through a network, however, each voice packet experiences a different latency, and hence, the inter-packet interval of the received voice traffic varies over time, referred to as *jitter*. In order to

TABLE 1
Parameters of Various VoIP Codecs

VoIP codec	Bit rate	T_{pkt}	P_{voice}	R_{max}
G.711	64 kb/s	20 ms	1280 bits	93.2
G.729a	8 kb/s	10 ms	80 bits	93.2
G.723.1	6.3 kb/s	30 ms	189 bits	93.2
G.722	64 kb/s	20 ms	1280 bits	129
AMR-WB	23.85 kb/s	20 ms	477 bits	129

compensate such jitters, a VoIP receiver uses a de-jitter buffer, which first enqueues a number of voice packets and then starts outputting them with a constant inter-packet interval. Then the VoIP packets are de-packetized, converted to an analog voice signal, and finally played back to a listener. In consequence, a voice signal experiences *mouth-to-ear delay*, which denotes the latency that a voice signal takes from a speaker's mouth to a listener's ear, consisting of packetization, processing, de-jitter buffer, wireline/wireless transmission delays, etc.

Fig. 1 illustrates how voice packets are generated, and what a mouth-to-ear delay consists of. It should be noted that each voice packet eventually experiences a fixed delay, which is the delay which the very first voice packet experiences, due to the de-jitter buffer. A voice packet experiencing an excessive delay, i.e., a delay longer than the delay bound, cannot be played back to the listener, and is equivalent to a packet loss. In this paper, we basically assume P-byte MAC frame payload size[1] with T_{pkt} ms inter-packet interval and the de-jitter buffer size of three packets.[2] We do not consider the silence suppression as it is known to deteriorate the quality of experience at the user side [24]. Accordingly, the packetization and de-jitter buffer delays are assumed to be T_{pkt} ms and $3T_{pkt}$ ms, respectively, and we ignore the processing delay which could be minimized depending on the computing power of VoIP devices. Table 1 summarizes the parameters of various codecs considered in this paper.

2.1.2 Performance Metric of VoIP

ITU-T Recommendation G.107 specifies E-model, which provides a convenient and objective quality metric, R-score. It is expressed by the following simplified equation [25]:

$$R = R_{max} - I_{delay} - I_{loss}, \qquad (1)$$

where R_{max} is the maximum R-score, and I_{delay} and I_{loss} represent impairment factors due to mouth-to-ear delay and packet losses of VoIP traffic, respectively [26], [27]. The values of these parameters are dependent on the codec. For example, the R_{max} values are presented for various codecs in Table 1. R-score greater than or equal to 80 is generally regarded as good voice quality. In this paper, we adopt E-model to evaluate the quality of VoIP traffic.

Fig. 2 plots the R-score of G.711 versus frame loss rate as varying mouth-to-ear delay. It is shown that the voice

---

1. This includes P_{voice}-byte voice data, 12-byte RTP header, 8-byte UDP header, 20-byte IP header, and 8-byte LLC/SNAP header, and represents the data size arriving at the MAC layer.

2. This implies that the very first voice packet gets out of the buffer when the fourth voice packet arrives at the buffer, and it is equivalent to $3T_{pkt}$ ms delay of the packets at the buffer if the packets arrive with a constant delay.

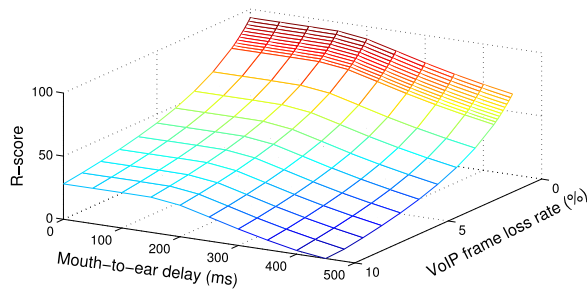Fig. 2. R-score of G.711 VoIP codec as a function of VoIP frame loss rate and mouth-to-ear delay.

Fig. 3. R-score of G.729a/723.1/722 and AMR-WB VoIP codecs in function of VoIP frame loss rate for 0 and 200 ms mouth-to-ear delay.

quality is mainly affected by the frame loss rate rather than the mouth-to-ear delay as long as the delay is under 150 ms; this trend remains the same for G.729a/723.1/722 and AMR-WB codecs as shown in Fig. 3. Since today's Internet can normally provide negligible packet losses and the latency less than 35 ms except for inter-continental communications [28], [29], [30], thus making it feasible to achieve the mouth-to-ear delay under 150 ms. We conclude that managing the packet losses over the wireless access link, i. e., WLAN, is very important to guarantee high QoS for VoWLAN, and leave the high-latency network environment as our future work.

2.2 Existing Rate Adaptation Algorithms

Today's IEEE 802.11 PHY standards provide various transmission rates with different throughput and error rate. Based on this multi-rate support, numerous researchers have been studying the *rate adaptation* of WLANs. We now overview state-of-the-art rate adaptation algorithms and their operations.

2.2.1 Automatic Rate Fallback (ARF) and Adaptive ARF (AARF)

Automatic rate fallback adopts a simple history-based rate adaptation algorithm [10]. Whenever N consecutive frames (referred to as *success threshold*, which is 10 frames by default) are successfully transmitted, the transmitter increases the PHY rate by one step. If consecutive frames (two frames by default) are lost, then the transmitter decreases the PHY rate by one step. Lacage et al. design adaptive ARF that adaptively manages the success threshold [11]. AARF mitigates the probing overhead by exponentially increasing the success threshold whenever a probing frame fails.

2.2.2 Collision-Aware Rate Adaptation (CARA)

In [12], Kim et al. propose collision-aware rate adaptation algorithm that can empirically distinguish the channel and collision errors. A WLAN station enables request-to-send (RTS) transmission after a frame transmission failure. Based on the transmission results of the RTS and the data frame following the RTS transmission, the WLAN station can tell whether the previous frame loss could be due to a channel error or collision, where the details can be found in [12]. CARA is further improved to quickly respond to the channel dynamics in CARA-RI [13], by modifying the resetting rule of the success counter.
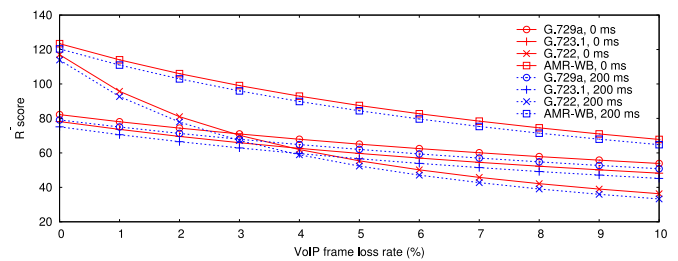
2.2.3 Robust Rate Adaptation Algorithm (RRAA)

Above-explained ARF-based rate adaptation algorithms, i. e., ARF, AARF, CARA, and CARA-RI, use the history of consecutive transmission trials. On the other hand, robust rate adaptation algorithm proposed in [14] facilitates frame error rate (FER) statistics. For each PHY rate, RRAA builds a table that contains (1) rate increasing/decreasing thresholds in terms of FER assuming 1,300 bytes frame length, and (2) estimation windows (*ewnd*) in terms of the number of frames. After transmitting as many as *ewnd* frames with a specific PHY rate, a WLAN station evaluates the FER and changes the PHY rate. RRAA also includes an adaptive usage of RTS (A-RTS).

2.2.4 SampleRate and Minstrel

SampleRate utilizes the FER statistics to select a PHY rate [15]. SampleRate estimates the expected transmission times including retransmissions with the given PHY rate and frame size. SampleRate then selects the PHY rate with the minimum expected transmission time, i.e., the PHY rate that maximizes the expected throughput. In order to collect an enough number of samples with various PHY rates, SampleRate probes a randomly selected PHY rate whenever a probing timer expires.

In an open-source WLAN device driver, MadWiFi [18], a revised version of SampleRate, called Minstrel [16], is implemented. Compared with SampleRate which does not specify the PHY rates that are used in retransmissions in its original proposal [15], Minstrel defines *retry chain* that specifies the PHY rates of retry frames based on the retransmission procedure of MadWiFi.

2.2.5 PHY-Assisted Rate Adaptations

PHY layer information can aid the selection of PHY rates. SoftRate [31] fetches bit error rate values, and AccuRate [32] captures symbol-level signal dispersions of constellation points from PHY. The weighted sum of multiple subcarriers' SNR values is used instead of packet-level SNR in [33]. These solutions utilize the receiver's feedback, which could be inaccurate considering the packetization interval of VoIP traffic, as to be discussed in Section 3.1. Moreover, they rely on the PHY information which might not be easy to access with most commercial WLAN devices, thus making them mostly impractical today.

2.3 IEEE 802.11 MAC and QoS Provisioning

The IEEE 802.11 MAC standard defines how WLAN stations share the wireless medium and deliver MAC frames to recipients. We briefly examine the MAC layer operation and QoS provisioning features of the IEEE 802.11 standard [8], and then discuss their influences on the QoS of VoWLAN.

2.3.1 Retransmission Policy of IEEE 802.11 MAC

The IEEE 802.11 enhanced distributed channel access (EDCA) MAC is based on the carrier sensing multiple access with collision avoidance (CSMA/CA) [8]. A WLAN station first senses the wireless medium to determine whether it is idle. If the medium has been idle for longer than arbitrary inter frame space (AIFS), then the WLAN station begins a *backoff* process with a backoff counter which is randomly chosen from [0, CW], where CW is a contention window size. Only after the medium remains idle until the backoff counter expires, the WLAN station transmits a frame. If the transmission is not acknowledged with an ACK frame, i.e., the frame is determined to be lost, then the WLAN station retransmits the previous frame after another backoff. Beginning from the minimum CW size (CWmin), the CW value is doubled for every frame transmission failure, until the CW reaches the maximum CW size (CWmax).

Continual retransmissions, however, could increase the frame loss rate in the following scenario. Suppose that a frame transmission fails due to a channel error. Depending on the randomly-chosen backoff counter value, a WLAN station has a chance to retransmit the frame within a short time interval, and such a retransmission is likely to fail again if the channel status remains bad and the PHY rate is the same. This problem becomes even worse for the VoWLAN with the differentiated service of the EDCA MAC as follows.

2.3.2 Access Category (AC) in IEEE 802.11

The 802.11 EDCA MAC provides differentiated services for real-time traffic types such as voice and video applications by classifying the types of traffic into four access categories, namely, voice (AC_VO), video (AC_VI), best effort (AC_BE), and background (AC_BK) [8]. The AC that requires high priority, e.g., AC_VO, has small AIFS/CWmin/CWmax values, so that the AC_VO traffic can more aggressively access the wireless medium than the other ACs.

Therefore, the time gap between two consecutive frame retransmissions of the VoIP traffic with the IEEE 802.11 EDCA MAC could be shorter than a few milliseconds. As shown in Fig. 4, however, the WLAN channel rarely changes in few milliseconds time interval so that the default retransmission policy of EDCA MAC with slow PHY rate adaptation is likely to fail again. We consequently conclude that we need not only a sophisticated PHY rate control for reducing the number of retransmissions, but also an appropriate retransmission scheduling for the reliable delivery of retry frames.

3 ENHANCING QoS OF VoWLAN

In this section, we discuss the limitations of the existing rate adaptation algorithms introduced in Section 2, and propose
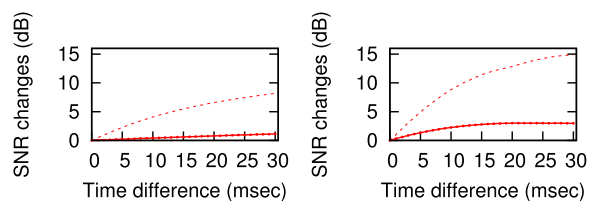
(a) Dopper velocity = 0.1 m/s.   (b) Doppler velocity = 1 m/s.

Fig. 4. Channel variation of Rayleigh fading model in NS-3 simulation; average (solid lines) and peak (dashed lines) SNR changes.

two novel features, namely, *fast decrease* and *retry scheduling* for VoWLAN. FD tries to reduce the number of retransmissions by controlling the PHY rates of retry frames, and RS postpones excessive retransmissions in order to overcome the deep fading of the wireless channel.

We in this paper assume IEEE 802.11n PHY with modulation and coding scheme (MCS) set from MCS 0 to MCS 7, i.e., 6.5, 13, 19.5, 26, 39, 52, 58.5, and 65 Mb/s. Note that the other higher PHY rates in 802.11n PHY, e. g., PHY rates with short guard interval, multi-input multi-output (MIMO), or channel bonding, have worse channel error performance [34]. Therefore those are not helpful for the QoS provisioning of VoIP traffic. Moreover, most of the today's smartphones, which are the main devices for VoIP service, have a single antenna, and hence, MIMO PHY rates are not available.

3.1 Limitations of the Existing Rate Adaptation Algorithms

ARF-based rate adaptation algorithms, i.e., ARF, AARF, CARA, and CARA-RI, necessitate the results of consecutive frame transmissions, and this is known to be the main reason of their slow response, which can be more problematic for VoWLAN. Suppose the AMR-WB voice codec with 20 ms packet generation interval. In order for a WLAN station to increase the PHY rate by one step, the WLAN station has to transmit 10 consecutive frames successfully, which might take longer than 200 ms including contentions and retransmissions. For the rate decreasing operation, the PHY rate is decreased by one step for every two consecutive failures, which might not provide robust PHY rates in the deep fading channel. Hence we need more agile rate increasing/decreasing rules.

Fig. 5a presents a simulation example (ns-3) of the PHY rate increasing/decreasing process of CARA-RI, which works similar to ARF. The solid line depicts the SNR variation, and O and X signs denote the frame transmission successes and failures, respectively, where the inner figure shows the detailed events in the deep fading near $t = 30.45$ seconds. Note that the deep fading channel can barely support 6.5 Mb/s PHY rate, while CARA-RI sequentially tries 26, 19.5, 13, and 6.5 Mb/s PHY rates, and eventually drops the frame due to the retry limit expiration.

PHY-assisted rate adaptation algorithms such as Soft-PHY and AccuRate work with the feedback of the receiver that estimates the current channel state and recommends the best PHY rate. Considering the packet generation interval of VoIP traffic, however, this feedback could be inaccurate at the transmitter side for the next VoIP frame
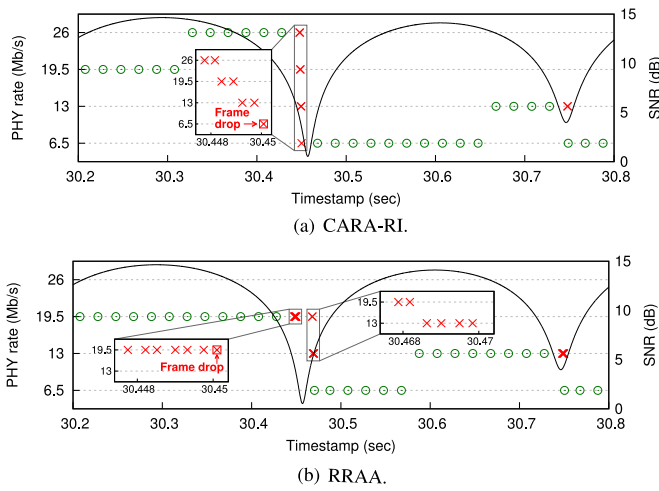
(a) CARA-RI.

(b) RRAA.

Fig. 5. The PHY rate increasing/decreasing process of CARA-RI and RRAA.

TABLE 2
PHY Rate Use of Fast Decrease

Transmission order	PHY rate	802.11n example
Initial try	Initial rate	58.5 Mb/s
1st retry	Unchanged	58.5 Mb/s
2nd retry	One step down	52 Mb/s
3rd retry	One step down	39 Mb/s
4th retry	Lowest [One step down]	6.5 Mb/s [26 Mb/s]
5th retry	Lowest [One step down]	6.5 Mb/s [19.5 Mb/s]
6th retry	Lowest [One step down]	6.5 Mb/s [13 Mb/s]

transmission. Hence we claim that the PHY-assisted algorithms are not feasible for VoWLAN. This problem due to the long frame generation interval of the VoIP traffic remains the same for RRAA.

In RRAA, the FER of the current PHY rate is estimated by transmitting as many as *ewnd* frames in order to exclude occasional frame losses and gather statistically-meaningful samples. Therefore, RRAA cannot be fast enough to trace the fast-fading channel, especially due to the *ewnd* sizes which range from six to 40 frames depending on the PHY rate [14]. Note that in the case of 39, 52, 58.5, and 65 Mb/s of 802.11n PHY (which correspond to 24, 36, 48, and 54 Mb/s of 802.11a PHY, respectively), the corresponding *ewnd* size is 40 frames. This is equivalent to approximately 1 second for the VoIP traffic with 20 ms packet generation interval; and it is too long time to adjust the PHY rate. Moreover, the default rate increase/decrease thresholds in [14] are designed for the data traffic with a 1,300 byte frame, and these thresholds are not valid for the short VoIP frames.

Fig. 5b demonstrates an example of the PHY rate selection of RRAA. The inner figure magnifies the consecutive transmission failures from $t = 30.43$ to 30.47 seconds. RRAA tries to steadily use 19.5 Mb/s PHY rate in the deep fading channel, and then the frame is dropped due to the expiration of the retry limit. Even after this frame drop, the same PHY rate is continually used for the next frame. This inefficiency of VoIP frame retransmissions can be alleviated by adopting a rate-decreasing rule for retransmissions as introduced in the following.

3.2 Rate Adaptation: Fast Decrease

Regarding the retransmission procedure of the IEEE 802.11 MAC, (re)transmission failures greater than the retry limit result in a frame drop. Therefore, it is highly recommended for the VoWLAN to retransmit using robust PHY rates, i.e., low PHY rates, in order to prevent VoIP frame drops and guarantee high voice quality.

We propose *fast decrease* that defines the PHY rates for the retransmissions of VoIP frames. FD basically decreases the PHY rate after the first retransmission failure, as the first frame loss could be due to both collision and channel errors;

note that decreasing the PHY rate on collision error deteriorates the performance of WLAN [12]. For instance, in an 802.11n PHY rate set from MCS 0 to MCS 7, if a frame is initially transmitted at 58.5 Mb/s PHY rate, then the first retransmission uses 58.5 Mb/s considering the possibility of collision errors, and the second and third retransmissions use 52 and 39 Mb/s, respectively, to cope with the channel error. In order to provide the best frame delivery probability, FD uses the lowest PHY rate beginning the fourth retransmissions, e.g., 6.5 Mb/s in the previous example. Table 2 summarizes the PHY rate use of FD with an example of 802.11n PHY MCS 0 to MCS 7, assuming retry limit of 7.

For ARF-based rate adaptation algorithms, FD needs to quickly increase the PHY rate after the lowest PHY rate transmission. Suppose that a frame is successfully delivered at the fourth retransmission using 6.5 Mb/s, assuming that the actual wireless channel can accommodate 26 Mb/s. If one selects 6.5 Mb/s for the next frame transmission, then it loses a chance to use higher PHY rates. Therefore, we define *reference rate* that is referred at the frame transmission after the successful retransmissions with the lowest PHY rate. According to the reference rate, one can try 26 Mb/s at the next frame transmission in the previous example as shown in Fig. 6. The reference rates are denoted within brackets in Table 2.

In the case of FD applied to RRAA, referred to as RRAA-FD, the transmission results of retries are reflected on the FER calculation as follows. The transmission failures of retries with low PHY rates are recorded as the failures of high PHY rates, while it is not the case for the transmission successes. For example, suppose that RRAA-FD initially fails to transmit at 52 Mb/s, and the fifth retransmission at 6.5 Mb/s by applying FD succeeds. RRAA-FD then assumes that if four retransmissions were using the initial 52 Mb/s, then all of those would fail, and hence, increases the failure counter of 52 Mb/s by 5, while its success counter remains unchanged.

Based on FD, we design *Agile ARF* (AgARF), which is an enhanced version of AARF. AgARF quickly decreases the PHY rate depending on the FD rule upon a frame transmission failure. In order to compensate the fast PHY rate decrease and opportunistically increase it back,
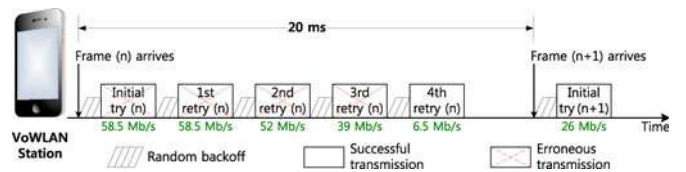

Fig. 6. An example of the FD design. PHY rates are denoted according to Table 2.

TABLE 3
PHY Rate Use of Fast Decrease with Retry Scheduling

Transmission order	PHY rate	802.11n example
Initial try	Initial rate	58.5 Mb/s
1st retry	Unchanged	58.5 Mb/s
2nd retry	One step down	52 Mb/s
3rd retry	Lowest [One step down(*)]	6.5 Mb/s [39 Mb/s]
4th retry	PHY rate of (*)	39 Mb/s
5th retry	Lowest [One step down]	6.5 Mb/s [26 Mb/s]
6th retry	Lowest [One step down]	6.5 Mb/s [19.5 Mb/s]

we reduce the minimum success threshold to 5 from 10, and resets the success counter in the same manner as CARA-RI. AgARF also adopts the A-RTS of RRAA and enables RTS transmissions, which could be effective in high-contention environments.

The transmissions with the lowest PHY rate in FD can reduce the transmission errors against bad channel status. Nonetheless, if the channel status becomes too bad due to the deep fading and/or severe hidden terminal interference, even the retransmissions with the lowest PHY rate may continue to fail. Hence, we develop a scheduling policy for the retransmissions in the following so that a WLAN station can overcome this problem.

3.3 Retransmissions: Retry Scheduling

If the retransmissions with the lowest PHY rate keep failing, then the VoWLAN station would better wait until the channel status becomes favorable, e.g., the wireless channel escapes the deep fading or the transmission of hidden terminal terminates. In this case, the question is until when the VoWLAN station waits for the retransmissions. Relying on the periodicity of the VoIP traffic, one possibility is postponing the retransmission until the next frame arrives at the MAC layer, so that the time diversity effect can be maximized while the next frame transmission is not much delayed by the delayed retransmissions of the preceding frame. As demonstrated in Fig. 4 and [35], [36], few tens of milliseconds is enough to result in 3∼4 dB variation of the WLAN channel quality that is required to change the PHY rate [31], [32], [37]. This also justifies delaying the retransmission with VoIP frame generation interval (10∼30 ms).

We refer to this delayed retransmissions as *retry scheduling*. More in detail, the VoWLAN station first attempts retransmissions up to three times with the PHY rates based on FD, while FD with RS is slightly modified so that the VoWLAN station uses the lowest PHY rate at the third retransmission. If the third retransmission fails, then the VoWLAN station postpones the fourth retransmission until the next VoIP frame arrives at the MAC layer. For the fourth retransmission after RS, i.e., the delayed retransmission, the VoWLAN station tries the reference rate of the third retransmission. Between the third and fourth retransmissions, the VoWLAN station may transmit other pending frames in order to prevent a queue blocking event. Table 3 summarizes the example of the PHY rate use of FD combined with RS including reference rates in brackets, and Fig. 7 presents the corresponding example.

Fig. 8 illustrates the operation of both FD and RS applied to CARA-RI from ns-3 simulation. The third retransmission of Frame n uses the lowest PHY rate and succeeds. In the case of Frame $(n+1)$, the initial transmission with the
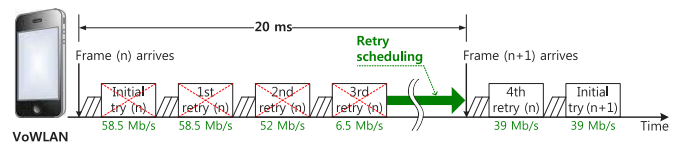

Fig. 7. An example of the RS design. PHY rates are denoted according to Table 3.

reference rate (19.5 Mb/s) fails, and the third retransmission with 6.5 Mb/s (where the reference rate becomes 6.5 Mb/s) fails as well. Hence, the fourth retransmission is delayed until Frame $(n+2)$ arrives. As the channel status gets better, the fourth retransmission of Frame $(n+1)$ with 6.5 Mb/s, which is the reference rate of Frame $(n+1)$, and the following initial transmission of Frame $(n+2)$ are successfully delivered.

One might argue that the proposed RS delays the delivery of a VoIP frame, thus degrading the QoS eventually. However, the extra delays of the VoIP frames can be absorbed by the de-jitter buffer. Note that with the proposed RS, a frame might experience an extra delay of about 20 ms, and the assumed de-jitter buffer (whose size is three packets) can mostly manage this extra delay. Moreover, just before the delayed retransmission, RS combined with FD tries the lowest PHY rate, which is supposed to be the most robust. The erroneous transmission at the lowest PHY rate implies that the other PHY rates are likely to fail too, and hence, a WLAN station has no reason to waste the wireless channel with erroneous transmissions; the wireless channel would rather be used for frames destined to or transmitted by other WLAN stations. Therefore, we claim that RS can enhance the QoS of VoIP traffic as well as the efficiency of the wireless channel usage.

Another technical issue of RS is its standard compliance. Strictly speaking, RS is not a standard-compliant operation since RS intentionally delays the frame retransmissions at the MAC layer. Note that according to the standard, frame retransmission instants are determined by the random backoff process and exponentially increasing CW values for retransmissions. However, it should be noted that RS does not incur any interoperability problem of existing VoWLAN devices as the medium access function via backoff processes is the same as the original EDCA MAC. As RS can substantially reduce the VoIP frame losses, we insist that RS is an effective operation for improving the QoS of VoWLAN.

As an extension of RS, the retransmissions can be delayed considering silence suppression, where the specific design is dependent on the codec and other delay components of the network, e.g., the wireline delay and de-jitter buffer size. The delayed retransmission frames could also
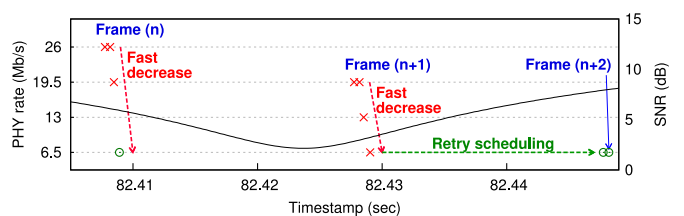

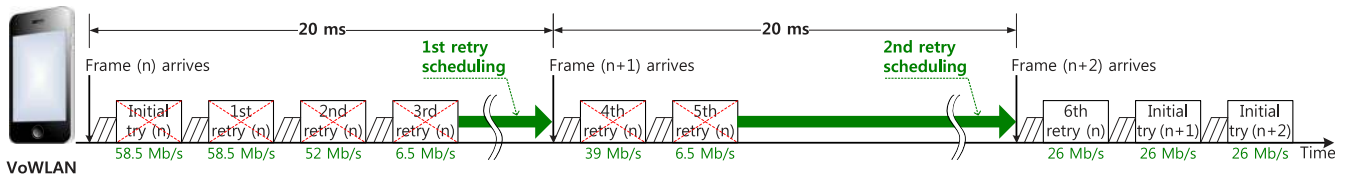Fig. 8. An example of FD and RS applied to CARA-RI.

Fig. 9. An example of the retry scheduling chain $(4, 2, 1)$. PHY rate usages are the same as in Table 3.

be transmitted back-to-back with the new frames by using frame aggregations defined in [38], in order to enhance the resource utilization, where we leave these issues as our future work. Another extension of RS is delaying the retransmissions further than the next packet arrival, i.e., more than one packet arrival period, as it is discussed in the following section.

3.4 Retry Scheduling Chain (RS Chain)

Suppose that the fifth retransmission of the proposed RS design in Table 3, which is transmitted with the lowest PHY rate, fails again due to deep fading. Relying on the rationale of RS, one can repeatedly delay the sixth retransmission until the next VoIP frame arrival in order to take advantage of the time diversity. We refer to the superposed RS operations as *retry scheduling chain*.

We denote the retransmission patterns of RS chain by using an m-dimensional vector, $(k_1, k_2, \ldots, k_m)$, where m represents the maximum number of VoIP frame intervals that are used to transmit one VoIP frame including RS operations, and k_i is the maximum number of packet transmission trials in the ith VoIP frame interval with constraint $\sum_i k_i =$ (retry limit). Then the RS design given in Section 3.3 is RS Chain $(4, 3)$, and Fig. 9 shows an example of RS Chain $(4, 2, 1)$. Note that in the second VoIP frame interval in Fig. 9, Frame $(n + 1)$ is not transmitted after the erroneous fifth retransmission of Frame n as the transmission of Frame $(n + 1)$ is likely to fail again.

The length of RS Chain, m, can be increased like RS Chain $(4, 1, 1, 1)$, so that VoWLAN stations can opportunistically improve the VoIP frame delivery ratio. However, we cannot indefinitely increase m due to the delay bound of VoIP frames. The maximum value of m should be less than $\lfloor \frac{T_{\mathrm{dj}}}{T_{\mathrm{pkt}}} \rfloor$, where T_{dj} and T_{pkt} denote the de-jitter buffer delay and packetization delay, respectively, in order to prevent the de-jitter buffer losses. As long as this condition holds, the de-jitter buffer can eliminate the delay intentionally added by RS Chain, as discussed in Section 3.3. In Section 4, we show that RS Chain $(4, 2, 1)$ achieves fairly good QoS for VoWLAN with an acceptable delay budget through simulation results.

4 PERFORMANCE EVALUATION: NS-3 SIMULATION

In this section, we evaluate the QoS of VoWLAN with various rate adaptation algorithms based on the ns-3 simulation results [17]. We basically assume G.711 codec without silence suppression, unless stated otherwise; the VoIP traffic is modeled as CBR traffic with 208 byte payload size and 20 ms packet generation interval. For G.729a/723.1/722 and AMR-WB codecs, voice data size and packet generation interval values are given in Table 1. We suppose the VoIP

communication scenario demonstrated in Fig. 1, where each VoIP terminal accesses the network through a WLAN AP connected to the Internet via a wireline backhaul, and the other VoIP terminal resides beyond the Internet. For the WLAN access, we assume IEEE 802.11n PHY with MCS indexes 0 to 7 and 802.11 EDCA MAC, where VoIP and TCP/UDP data traffic are mapped into the access category AC_VO and AC_BE, respectively, with default channel access parameters given in [8]. The delays of the wireline Internet and de-jitter buffer are assumed to be 35 and 60 ms, respectively [28], [29], [30]. For each set of simulation scenarios, we average the results of 50 simulation runs where each runs for 100 second simulation time, and the R-score of VoIP sessions. We also adopt the path-loss model with path-loss exponent of 4 for a typical indoor environment, and the Rayleigh fading channel model with the Doppler velocity of 0.1 m/s for static stations and 1 m/s for moving stations, respectively.

4.1 Comparing Algorithms

We compare the performance of the following well-known rate adaptation algorithms, while some of those are modified for VoWLAN support as explained below.

4.1.1 ARF-Based Families

We basically consider ARF and AgARF, and the revision of AgARF by using RS (AgARF w/RS). CARA-RI is also revised by using FD (CARA-RI-FD), and both FD and RS (CARA-RI-FD w/RS).

4.1.2 FER-Statistics-Based Families

Various versions of RRAA are considered. Basically the original RRAA adopts the FER table which is calculated assuming 1,300 bytes frame payload size and 40 *ewnd* size, which are not suitable for the VoIP traffic. Hence, we modify the FER table using VoIP frame payload size and use the *ewnd* size of 10, and then apply FD (RRAA-e10-FD). Finally RRAA-e10-FD is further revised with RS (RRAA-e10-FD w/ RS). Minstrel, another representative FER-statistics-based scheme, is also compared.

4.1.3 Reference Algorithms

We implement a reference rate adaptation algorithm, referred to as *Genie*, that ideally determines the best PHY rate without any overhead. Specifically, Genie selects the highest PHY rate that yields FER less than 1 percent for a given wireless channel based on the SNR. We also consider the 6.5 Mb/s fixed rate case as a baseline reference because 6.5 Mb/s is the most robust PHY rate, which can potentially maximize the QoS of VoWLAN.
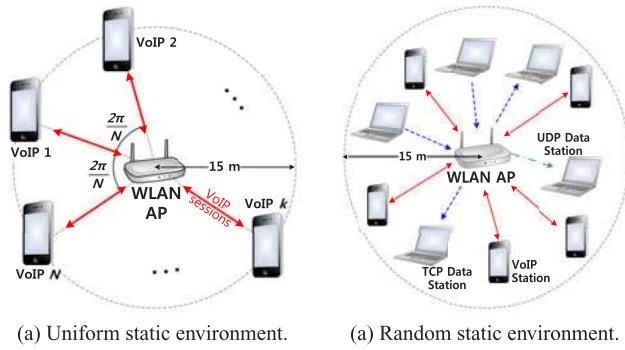
(a) Uniform static environment.   (a) Random static environment.

Fig. 10. Simulation scenarios for the static network topology of WLAN.

4.2 Static Topology

4.2.1 Fast Decrease and Retry Scheduling

First, we study the QoS provisioning capabilities of the existing rate adaptation algorithms in a static environment. We consider a WLAN composed of an AP and N static VoIP stations, where the stations are equally and uniformly apart from the AP by 15 m, as shown in Fig. 10a. At this distance, the time-varying wireless channel supports 26 and 39 Mb/s PHY rates in average. We present the lowest R-score out of $2N$ VoIP sessions (including uplink and downlink sessions) in the following figures, so that we can check whether all the VoIP sessions can be provided a satisfactory R-score.

Fig. 11 demonstrates the QoS of ARF, CARA-RI, RRAA, Minstrel, 6.5 Mb/s fixed rate, and Genie rate adaptation algorithms. The upper histogram shows the R-score values, and the lower one shows the average channel occupancy time used to transmit a single VoIP frame including ACK (and RTS/CTS for RRAA and CARA-RI) overheads with various rate adaptation algorithms. We verify that with small numbers of VoIP stations, RRAA performs even worse than ARF due to the frame drops caused by its slow adaptation and inappropriate FER table reference; RRAA steadily uses high PHY rates so that the retransmissions could fail frequently. On the other hand, ARF and CARA-RI decrease the PHY rate by one step after two consecutive failures, and hence, the retransmissions have higher probabilities to be successfully delivered.

One might argue that using the lowest PHY rate in a fixed manner, i.e., the 6.5 Mb/s case, provides the best
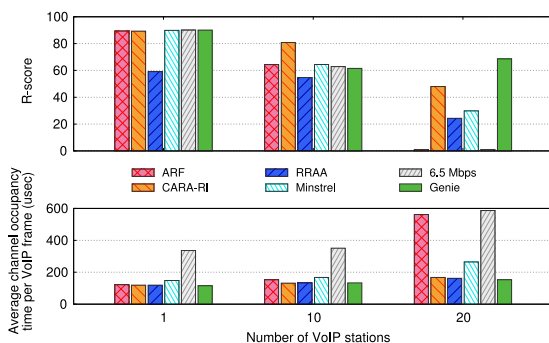

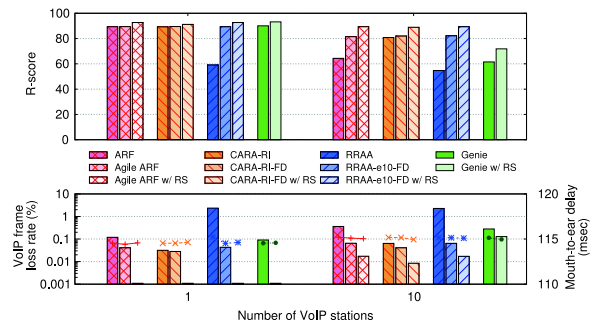Fig. 11. The R-score and channel occupancy time of various rate adaptation algorithms.

Fig. 12. The R-score (upper bars), VoIP frame loss rate (lower bars), and mouth-to-ear delay (lower marks) of the revised rate adaptation algorithms using fast decrease (denoted as -FD) and retry scheduling (denoted as w/RS).

QoS thanks to its robustness. This is true when there exists only a single station; the R-score of 6.5 Mb/s is the best. However, as shown in Fig. 11, it is not the case for the high contention environment with 20 VoIP stations as the lowest PHY rate consumes too much of the wireless channel, so none of the stations can be properly supported in this environment. It is also known that ARF mostly uses the lowest PHY rate in high contention environments [12], and we confirm that by observing the R-score and channel occupancy time of ARF and 6.5 Mb/s. Accordingly, the R-score of ARF with 20 VoIP stations is as low as that of 6.5 Mb/s case, and we reconfirm that it is required to mitigate the collision problem by using RTS adaptively as CARA-RI and RRAA do.

Now we confirm the performance gain of the FD and RS designs in Fig. 12. Due to the lack of space, we only present ARF, CARA-RI, RRAA, and Genie rate adaptation algorithms with one and ten VoIP stations; for the 20 VoIP stations case, none of these rate adaptation algorithms can support R-score greater than 80, which is out of our interest. All the above schemes are revised by applying (1) FD or (2) both FD and RS. The R-score shown in the upper figure is enhanced by applying FD, while RS further enhances the R-score above 80. The only exception is Genie, which does not use RTS. Especially for RRAA, the performance gain of FD is significant since the original rate selection of RRAA is too static. RS improves the R-score for all of the rate adaptation algorithms by removing the continuous retransmission failures with the 6.5 Mb/s PHY rate and reducing the VoIP frame loss rate to below 0.1 percent. We also observe that FD and RS algorithms rarely affects the mouth-to-ear delay of VoIP frames thanks to the de-jitter buffer operation. From Fig. 12, we conclude that the FD and RS features improve the existing rate adaptation algorithms, where the gain varies from 10 percent (in the case of CARA-RI) to 63 percent (in the case of RRAA). In the following discussions, we present ARF and RRAA, as well as their variants, as representative schemes of ARF-based families and FER statistics-based families, respectively.

4.2.2 Retry Scheduling Chain

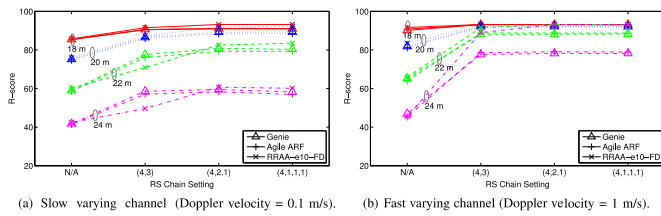Now, we evaluate the performance of RS Chain depending on the chain length. We consider a bidirectional one-to-one

(a) Slow varying channel (Doppler velocity = 0.1 m/s). (b) Fast varying channel (Doppler velocity = 1 m/s).

Fig. 13. R-score of various RS Chain settings depending on the channel variation and link distance.

Fig. 14. The empirical CDF of the R-scores of various rate adaptation algorithms. The table inside presents the average R-scores.

G.711 VoIP communication, where two VoWLAN nodes are 18~24 m apart from each other. The wireless channel is assumed to vary with Doppler velocity of 0.1 and 1 m/s representing slow and fast varying channels, respectively. We compare three different RS Chain settings, $(4, 3)$, $(4, 2, 1)$, and $(4, 2, 1, 1)$, and their de-jitter buffers are set to 60, 80, and 100 ms, respectively, in order to prevent the de-jitter buffer losses due to RS Chain.

Figs. 13a and 13b show the R-score of Agile ARF, RRAA-e10-FD, and Genie rate adaptation algorithms in slow and fast varying channel environments, respectively. At 22 m distance in slow varying channel, which supports 6.5~13 Mb/s PHY rates in average, the rate adaptation algorithms without RS Chain that do not include any delayed retransmission cannot perform well due to the consecutive failures of the lowest PHY rate transmissions. By employing RS Chain, however, the VoWLAN stations can opportunistically deliver the VoIP packet, leveraging the time diversity of the wireless channel. The gain of RS Chain is more significant in fast varying channel as shown in Fig. 13b, but the gain of RS Chain $(4, 1, 1, 1)$ against RS Chain $(4, 2, 1)$ is marginal in both slow and fast varying channel.

We demonstrate the simulation results with a single VoWLAN pair in this scenario. Note that, however, the gain of RS diminishes as the number of contending stations increases due to the medium contention; the VoWLAN stations do not need to intentionally delay the retransmissions because those are eventually delayed during the contention. Therefore we conclude that delaying the retransmissions up to two or three VoIP packet generation intervals, as in RS Chain $(4, 3)$ or $(4, 2, 1)$, is good enough to improve the QoS of VoWLAN.

The detailed optimization of RS Chain should depend on the voice codec (VoIP packet generation interval), network delay, and de-jitter buffer size. Specifically, in order not to induce additional delay due to RS Chain, the length of RS Chain multiplied by VoIP packet interval should be less than the de-jitter buffer delay; meanwhile, the sum of de-jitter buffer delay and network delay should be less than the VoIP delay bound (which depends on the voice codec) to prevent the VoIP packet from being dropped at the de-jitter buffer. We plan to study this issue in our future work. In the following evaluations, we refer to RS Chain $(4, 3)$ as RS.

4.3 Random Topology with Heterogeneous Traffic Types

In this scenario, five static VoIP stations and five static data stations are randomly deployed within a circle of 15 m radius, where an AP resides at the center of the circle as depict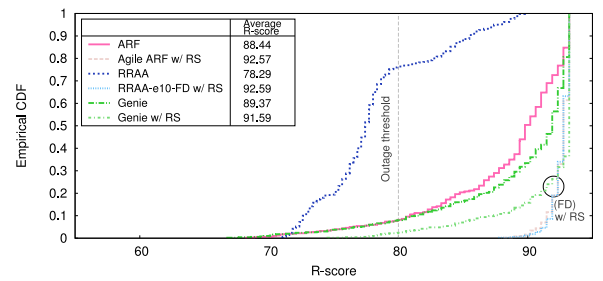ed in Fig. 10b. The numbers of TCP downlink, TCP uplink, UDP downlink, UDP uplink stations are two, one, one, and one, respectively, as downlink TCP traffic is more general in practical WLAN environments, where each TCP/UDP flow has 2 Mb/s traffic load from the application layer with 1,508 byte frame payload size. Both VoIP and data stations use the same rate adaptation algorithm, except the RS feature which is used only by VoIP stations.

Fig. 14 demonstrates the empirical cumulative distribution functions (CDFs) of the R-scores of ARF, RRAA, and Genie rate adaptation algorithms including their variants. An interesting observation is that the average R-score of this scenario with both VoIP and TCP data stations is slightly higher than that of the previous scenario with only VoIP stations. This is due to the random topology which has high chances that VoIP stations get closer to the AP, as well as the advantage of the time diversity effect in retransmissions of the VoIP frames. That is, the retransmissions with the same PHY rate can succeed after the channel contention with the data stations, and the diversity effect of the heterogeneous network is more advantageous than in the VoIP-only case since data transmissions take longer time than short VoIP frames.

From Fig. 14, it is shown that RRAA cannot support 76 percent of VoIP stations with the R-score higher than 80. The outage probabilities, i.e., the probabilities that the R-score is below 80, of ARF and Genie are 10 percent. By applying FD and RS features, however, Agile ARF with RS and RRAA-e10-FD with RS can reduce the outage probability to zero. Genie with RS performs worse than those due to the lack of adaptive RTS algorithm; note that in this heterogeneous random topology, hidden stations deteriorate the R-score of VoIP especially due to the UDP data stations. We discuss the effect of hidden stations in Section 4.5 more in detail.

4.4 Mobile Topology

A rate adaptation is required to be highly responsive in mobile environments in order to track the channel dynamics. We now evaluate the QoS of the rate adaptation algorithms considering their responsiveness in the following mobile scenario.

4.4.1 Straight-Line Scenario

We consider a simple mobile scenario with a single VoIP station which moves back-and-forth along a straight line at 1 m/s velocity as shown in Fig. 15a. Once the VoIP station reaches the end of the line, where the VoIP station can marginally use the 6.5 Mb/s PHY rate, it turns around and goes
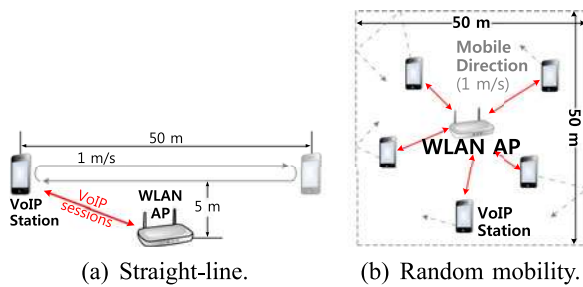
(a) Straight-line.   (b) Random mobility.

Fig. 15. Simulation scenarios for the WLAN with mobile stations.

back. We evaluate the R-scores every 5 seconds and record them as a function of time.

Fig. 16 shows the time history of the R-score performance in this mobile scenario. Note that ARF that cannot quickly adapt to the channel dynamics performs badly when the VoIP station moves far from the AP, and RRAA does even worse. This goes the same for Genie as well, since the 6.5 Mb/s transmissions could fail due to the deep fading. On the other hand, the rate adaptation algorithms revised by using the FD and RS features dramatically improve the R-score performance, since using the robust PHY rate as well as delaying the retransmissions can increase the frame delivery ratio in long-distance communications.

4.4.2 Random Mobility Scenario

We now randomly deploy five VoIP stations in a square of 50 m width, and let the WLAN stations move to random directions at 1 m/s velocity (Fig. 15b). When a WLAN station faces the boundary of the square, the WLAN station randomly changes its direction and keeps moving. Thanks to the lowest PHY rate use for retransmissions as well as delayed retransmissions to avoid the deep fading, the revised Agile ARF w/RS, RRAA-e10-FD w/RS, and Genie w/RS can achieve 91.97, 92.24, and 90.67 R-scores, respectively, while their original versions, i.e., ARF, RRAA, and Genie have 70.81, 57.71, and 76.66 R-scores, respectively.

4.5 Hidden Terminal Environment

Now we investigate the performance of the rate adaptation algorithms in the WLAN environments with hidden terminals. Fig. 17a illustrates the experimental deployment of a VoIP station and a TCP data station in opposite directions. All the WLAN stations use the same rate adaptation algorithm, except the RS feature which is only applied for the VoIP traffic. Since the WLAN stations are far enough, these two cannot sense the transmission of
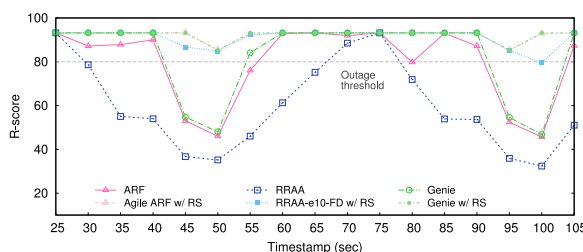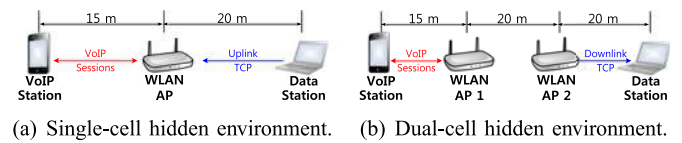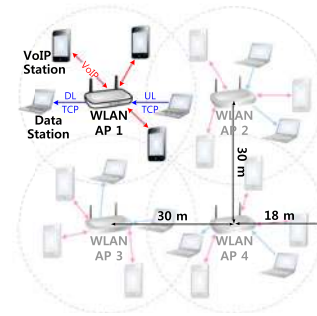

(a) Single-cell hidden environment.   (b) Dual-cell hidden environment.

(c) Multi-cell hidden environment

Fig. 17. Simulation scenarios for the WLAN with hidden terminals.

each other. The TCP data station generates an uplink TCP flow with 1 Mb/s traffic load. Fig. 17b illustrates the case that hidden terminals are associated with two different APs. Two APs can sense the transmission of each other, but the WLAN stations cannot sense their neighboring AP that they are not associated with. The TCP data station receives a downlink TCP flow with 1 Mb/s traffic load. Finally, Fig. 17c shows the hidden terminal topology with four APs. Each AP accommodates three VoIP stations, one downlink TCP station, and one uplink TCP station that are randomly deployed in 18 m cell radius. Inter-AP distance is 30 m, so that four WLAN cells partially overlap with each other. Each TCP flow has 1 Mb/s traffic load.

The R-score performances of uplink VoIP sessions in hidden terminal environments are presented in Fig. 18 when both VoIP and TCP data stations use the same rate adaptation algorithm. We omit downlink R-scores as those are higher than uplink ones; due to the settings of hidden terminals, the uplink VoIP session suffers from the hidden terminal interference. In the case of RRAA, the hidden terminal problem can be resolved via adaptive RTS transmissions, but the slow PHY rate adaptation of RRAA deteriorates both uplink R-scores close to 60. By applying adaptive RTS and FD/RS algorithms, Agile ARF and RRAA-e10-FD enhance the R-score to over 80. It should be also noted that Genie with RS can overcome the hidden terminal problem without RTS transmissions thanks to the delayed retransmission. We therefore conclude that the proposed FD and RS algorithms can

Fig. 16. The R-score records during 80 seconds moving along the straight line. The VoIP station is the closest to the AP at 25 and 75 seconds.
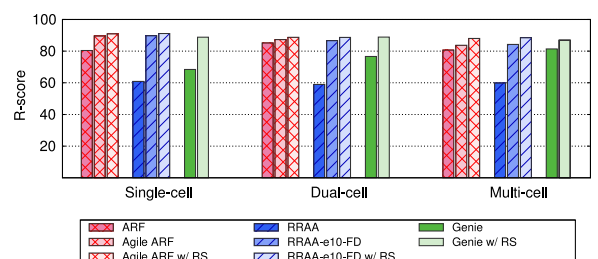

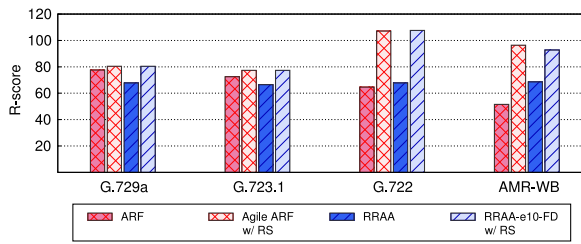Fig. 18. The R-score in hidden terminal environments.

Fig. 19. The R-score of various VoIP codecs.

improve the QoS provisioning for VoWLAN in hidden terminal environments.

4.6 Effects on Various VoIP Codecs

We finally evaluate the performance gain of FD and RS algorithms on G.729a/723.1/722 and AMR-WB codecs. The simulation topology is shown in Fig. 10a, where there are 10 VoIP stations, and the cell radius is determined depending on the VoIP codec so that approximately 67 R-score is achieved by RRAA.

Fig. 19 demonstrates the R-score of ARF, RRAA, and their variants for the four codecs. We observe that the revised Agile ARF w/ RS and RRAA-e10-FD w/ RS achieve the R-score over 80. The only exception is G.723.1, which has the maximum R-score of 78.2 due to its codec impairment; in this case, FD and RS algorithms enhance the R-score close to the maximum. We also observe the drastic performance gains of FD and RS algorithms for G.722 and AMR-WB codecs. This is due to their loss-sensitiveness as shown in Fig. 3. Therefore, we conclude that the QoS enhancements of FD and RS algorithms are valid irrespective of the VoIP codec.

5 PERFORMANCE EVALUATION: MADWIFI IMPLEMENTATION

We have implemented the proposed Agile ARF with RS in MadWiFi 0.9.4 using Lenovo X41 laptop equipped with Ubuntu 7.0.4 and Cisco Air-CB21AG WLAN card with 802.11a PHY. Due to the limitations of MadWiFi, we have the following modifications of the proposed Agile ARF algorithm.

5.1 Implementation of Adaptive RTS

When MadWiFi driver delivers a packet from TCP/IP layer to hardware abstraction layer (HAL) that controls the WLAN network interface card (WNIC), the packet, i.e., MAC service data unit (MSDU) is accompanied with *transmit descriptor* (or packet descriptor) generated by MadWiFi. The transmit descriptor is used to specify relevant information for the packet transmission, e.g., packet length, transmission power and PHY rate, retry limit, whether to use RTS/CTS for this packet.

Multi rate retry (MRR) refers to the MadWiFi feature that supports controlling the PHY rates and number of retransmissions by using the transmit descriptor. The MRR is specified by a vector $(R_1, C_1, R_2, C_2, R_3, C_3, R_4, C_4)$; the frame is first transmitted at PHY rate R_1 up to C_1 trials (including retransmissions). If the frame is not acknowledged after C_1 transmission attempts, then the frame is retransmitted at

PHY rate R_2 up to C_2 trials, and so on. If the frame is not acknowledged after $(C_1 + C_2 + C_3 + C_4)$ transmission attempts, then the frame is discarded. By using the MRR, we can easily implement *fast decrease* in MadWiFi. Specifically, assuming the initial PHY rate R, Agile ARF has transmit descriptor of $(R, 2, R^-, 1, R^{--}, 1, R_0, 3)$, where R^- is the PHY rate that is one step lower than R, and R_0 is the lowest PHY rate. The evolution of R follows the algorithm explained in Section 3.2.

It should be noted that the use of RTS/CTS is predetermined in the transmit descriptor which also defines retransmissions. Hence we cannot enable (or disable) RTS/CTS during the frame retransmission procedure; once RTS/CTS is enabled (disabled) at the initial frame transmission, the following frame retransmissions shall also include (exclude) RTS/CTS transmissions. Due to this limitation, A-RTS of RRAA that dynamically toggles the RTS/CTS usage for the frame retransmissions by managing RTSWnd and RTSCounter parameters for each MAC protocol data unit (MPDU) (re)transmission cannot be implemented in MadWiFi. Instead, we here employ *implementable RTS* (iRTS) algorithm,[3] which is a modified version of A-RTS in RRAA, so that it can be implemented in MadWiFi driver. The iRTS design basically adopts RTSWnd and RTSCounter parameters of A-RTS, but additionally considers the transmission descriptor of MadWifi. Different from A-RTS of RRAA, RTSWnd and RTSCounter of iRTS are updated after an MSDU transmission by referring to the transmission results of HAL as follows.

iRTS has two operating parameters, RTSWnd and RTSCoutner, and one decision parameter x, which is the number of retransmissions of the previous MSDU, where x is reported by HAL after an MSDU transmission.[4] We also define a function $R(x)$ that denotes the transmission rate used for the xth retransmission, where $R(0)$ is the initial transmission rate. If $R(x)$ is greater than or equal to $R(0)$ where $x > 0$[5] and RTS is not used for the previous MSDU transmission, then RTSWnd increases by one. This is based on the uncertainty of frame losses. Note that consecutive (re)transmission failures with the same transmission rate could be due to either collision or channel error. However, a successful retransmission with the same transmission rate suggests that the previous (re)transmission failures are due likely to collisions, assuming that the retransmissions occur in the channel coherence time of the WLAN [35], [36]. Therefore, using RTS for the next MSDU transmission could be helpful. Meanwhile, if $x > 0$ with RTS or $x = 0$ without RTS, then RTSWnd is halved as RTS is likely not to be helpful. In other cases, RTSWnd remains the same. Algorithm 1 presents the procedure explained above.

---

3. An initial version of the algorithm presented in this paper was presented in [39].

4. $x = 0$ denotes that the previous MSDU transmission did not include any retransmissions, i.e., the MSDU is successfully transmitted at the initial transmission.

5. $R(x) > R(0)$ for $x > 0$ is unlikely, but can occur with some rate adaptation algorithms such as Minstrel that probes randomly selected rate which could be higher than the initial rate [16].

5.2 Implementation of Retry Scheduling

In order to implement *retry scheduling*,[6] we have revised MadWiFi so that every packet that arrives at the MAC from TCP/IP layer is basically copied and stored in local memory, where the original packet has the transmit descriptor $(R, 2, R^-, 1, R_0, 1, R_0, 0)$ and the copy has $(R^{--}, 1, R_0, 2, R_0, 0, R_0, 0)$.[7] At the very beginning, MadWiFi first delivers the original packet to HAL. Being responded by the transmission result of the original packet, MadWiFi removes the copied packet if the original packet's transmission is successful, or maintains it otherwise.

Algorithm 1 iRTS Algorithm: w/o RS chain

1: RTSWnd = RTSCounter = 0;
2: **while** true **do**
3:    x = report(# of retransmissions);
4:    **if** $x > 0$ and !RTSOn and $R(x) \geq R(0)$ **then**
5:      RTSWnd++;
6:      RTSCounter=RTSWnd;
7:    **else if** ($x == 0$ and !RTSOn) or ($x > 0$ and RTSOn) **then**
8:      RTSWnd = RTSWnd/2;
9:    **end if**
10:   **if** RTSCounter $>$ RTSWnd **then**
11:     RTSWnd = RTSCounter;
12:   **end if**
13:   **if** RTSCounter > 0 **then**
14:     Enable RTS(next_frame_descriptor);
15:     RTSCounter−−;
16:   **end if**
17: **end while**

Then whenever MadWiFi receives a new packet from TCP/IP layer, MadWiFi checks whether the copy of the previous packet remains at the local memory. If the memory is not empty, then the copy of the previous packet and the newly arrived packet are delivered to HAL back-to-back, and the memory is substituted by the copy of the newly arrived packet. If the memory is empty, then only the newly arrived packet is delivered to HAL and MadWiFi copies it at the memory. In summary, we have implemented the proposed Agile ARF with iRTS in MadWiFi including fast decrease and retry scheduling by using MRR and additional memory block in our testbed.

It should be noted that this retry scheduling implementation leverages two transmit descriptors for an MSDU transmission. Hence, MadWiFi driver has more chances to enable/disable RTS/CTS during the frame retransmissions, compared with the case in Section 5.1. That is, when MadWiFi driver tosses the copy of the previous packet to HAL after the retry scheduling, its transmit descriptor can be modified in order to toggle the RTS/CTS transmission. Different from the RTSWnd and RTSCounter updates on MSDU transmissions in Section 5.1, retry scheduling makes those can be updated after a group of MPDU transmissions, which is determined by the settings of retry scheduling chain. Algorithm 2 demonstrates the modified iRTS procedure considering retry scheduling chain (4, 3).

---

6. We here describe the implementation of retry scheduling chain (4, 3), while it can be easily extended to an arbitrary retry scheduling chain.

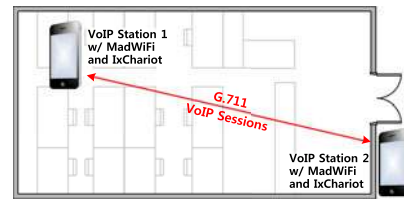7. We assume that the initial PHY rate is R, and R^{--} is the PHY rate two steps lower than R.

Fig. 20. Static experimental environment of MadWiFi testbed.

Algorithm 2 iRTS Algorithm: w/ RS chain

1: RTSWnd = RTSCounter = 0;
2: **while** true **do**
3:    x = report(# of retransmissions);
4:    **if** ($x == 1$ and !RTSOn and $R(x) \geq R(0)$) or ($x == 3$ and RSEnabled and $R(x) \geq R(0)$) **then**
5:      RTSWnd++;
6:      RTSCounter=RTSWnd;
7:    **else if** ($x == 0$ and !RTSOn) or ($0 < x < 3$ and RTSOn) or ($x == 3$ and !RSEnabled and RTSOn) **then**
8:      RTSWnd = RTSWnd/2;
9:    **end if**
10:   **if** RTSCounter $>$ RTSWnd **then**
11:     RTSWnd = RTSCounter;
12:   **end if**
13:   **if** RTSCounter > 0 **then**
14:     Enable RTS(next_frame_descriptor);
15:     RTSCounter−−;
16:   **end if**
17: **end while**

5.3 Experimental Results

We compare the performance of the implemented Agile ARF against ARF, adaptive multi rate retry (AMRR), ONOE, and Minstrel [10], [11], [16], [18]. AMRR is a customized version of AARF for the MadWiFi implementation, and ONOE basically selects the highest PHY rate that yields the frame loss rate less than 50 percent. AMRR, ONOE, and Minstrel are already implemented in MadWiFi, and we additionally implement ARF for the experiment.

We use IxChariot for VoIP traffic generation as well as the performance measurement tool [40], and AirPcap NX for packet sniffing [41]. The bidirectional VoIP call lasts for 150 seconds with G.711 voice codec and 20 ms packet generation interval.

5.3.1 Static Environment

Fig. 20 shows the static experimental environment in our laboratory, which is a typical office room with a pair of VoIP stations. One VoIP station resides in the room, and the other does out of the entrance in order to test relatively poor channel state. Though the VoIP stations do not move, people are walking around and doors are frequently open and closed, so that the channel highly fluctuates. For a fair comparison over the time-varying channel, we install Agile ARF in one VoIP station and other comparing rate adaption algorithm in the other station, and then generate the bidirectional VoIP traffic.

Fig. 21 presents the performance of Agile ARF and the other comparing algorithms in MadWiFi experiment, with error-bars representing the minimum and maximum values. Each result is obtained by averaging five runs for each set of experiments. We observe that the performance of Agile ARF differs from each set of experiments due to the
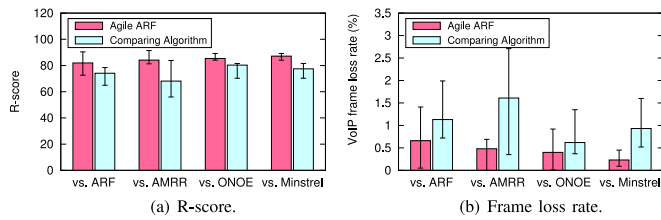
Fig. 21. Performance of Agile ARF against comparing algorithms in MadWiFi experiment.

highly fluctuating channel, since we cannot reproduce exactly the same channel in each experimental set. However, Agile ARF is shown to achieve consistently higher R-score than the other algorithms from 6.4 percent (versus ONOE) to 23.5 percent (versus AMRR) by reducing the VoIP packet loss rates from 35.8 percent (versus ONOE) to 75.14 percent (versus Minstrel), thanks to fast decrease and retry scheduling.

We investigate the R-score history of Agile ARF and Minstrel. Fig. 22 plots the R-score history every 3 seconds. Though the R-score of Minstrel fluctuates in lightly fading channels (shaded region R1), Agile ARF performs relatively stable. Agile ARF also shows better performance in deep fading channels (shaded region R2); the R-score drops less and quickly restores compared with Minstrel. Through the testbed experiments, we confirm the performance gain of the proposed Agile ARF as well as its feasibility for implementation in actual WLAN devices.

5.3.2 Mobile Environment with Heterogeneous Traffic

We evaluate the performance of the proposed algorithm in indoor mobile environment with heterogeneous types of traffic, as shown in Fig. 23. We deploy a Cisco Aironet 1040 AP at the entrance of our laboratory, and four VoIP stations (P1~P4) implemented via MadWiFi in Lenovo laptops at the same floor, where the VoIP station at P3 traverses along the hall at approximately 1 m/s walking speed. We also place a TCP station (P5) that transmits fully-loaded uplink TCP traffic. Note that the VoIP stations at P1 is a hidden interferer to all the other stations including the TCP station due to the experimental topology. All the VoIP stations are transmitting and receiving bi-directional G.711 VoIP sessions with 20 ms packet generation interval for 150 seconds. We compare ARF, Minstrel, and AgARF w/ RS rate adaptation algorithms. AgARF w/RS basically does not use RTS at all, while it is further modified by implementing the iRTS algorithm in Section 5.2 (AgARF w/ RS & iRTS), or always use the RTS (AgARF w/RS & RTS).
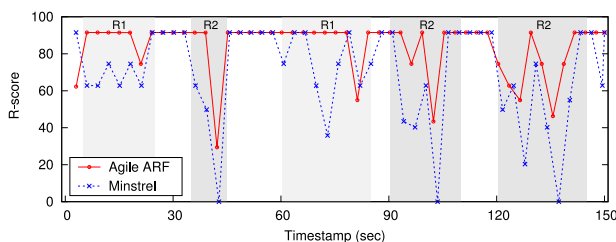

Fig. 22. Evolution of R-score of Agile ARF and Minstrel. Regions R1 and R2 depict light and deep fading channels, respectively.
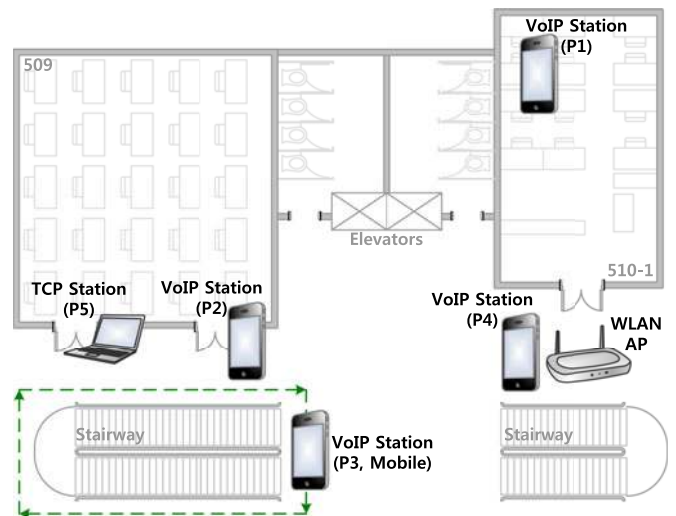

Fig. 23. Mobile experimental environment of MadWiFi testbed.

We present the uplink R-scores of four VoIP stations in Fig. 24 including their maximum and minimum values with error-bars. As there is no hidden terminal in downlink VoIP transmissions, the downlink R-scores of all four stations are above 90; therefore we omit the downlink R-score in Fig. 24. ARF, Minstrel, and Agile ARF w/RS do not use of RTS/CTS transmissions, so that P1 which suffers from severe hidden interference from all the other stations has poor R-score. This is the same for P3 because P3 cannot sense P1 as well as P4 when P3 is far away from AP. By enabling RTS transmissions, however, we see that the R-scores of P1 and P3 are remarkably improved over 80. We also observe that the R-score of the proposed iRTS algorithm is very close to the case that VoIP stations always use RTS, which has large RTS transmission overhead. Meanwhile, as iRTS adaptively manages the use of RTS, so that the channel occupancy time of iRTS is less than RTS. As a result, the TCP station with iRTS in Fig. 23 achieves better throughput than that with RTS.

6 CONCLUSION AND FUTURE WORK

In this paper, we investigate the characteristics of VoWLAN systems, and discuss the pros and cons of the existing rate adaptation algorithms regarding the VoIP traffic. Based on the loss-sensitiveness of the VoIP traffic, we design two novel features for the rate adaptation of VoWLANs. Fast decrease guides the PHY rates of retransmission frames to
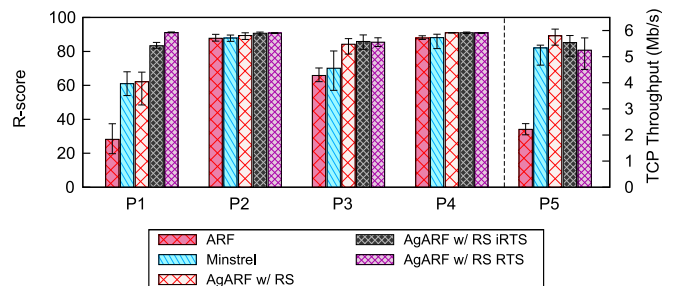

Fig. 24. Uplink R-scores of four VoIP stations at P1~P4 and TCP throughput at P5. Downlink R-scores are above 90 for all the stations as there is no hidden terminal.

low(est) PHY rates so that those frames can be successfully delivered before the retry counter expires. Retry scheduling manages the timing for the retransmissions in order to avoid both deep channel fading and hidden terminal interference. Through extensive ns-3 simulations and MadWiFi measurements, we demonstrate that applying these two features to the existing rate adaptation algorithms can dramatically improve the QoS of VoWLAN.

We are currently considering the extension of the proposed approaches for real-time video traffic, whose QoS is also very sensitive to its packet loss rate. Moreover, through the future work on refining the RS design, i.e., jointly optimizing the delayed transmission and de-jitter buffer size considering silence suppression and high-latency network, we expect to further enhance the QoS of VoWLAN.

ACKNOWLEDGMENTS

REFERENCES

[1] B. Kim et al., "Enhancing QoS of Voice over WLANs," *Proc. IEEE Int'l Symp. a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, June 2012.
[2] Wi-Fi Alliance, http://www.wi-fi.org, online link, 2014.
[3] Y. Xiao et al., "Protection and Guarantee for Voice and Video Traffic in IEEE 802.11e Wireless LANs," *Proc. IEEE INFOCOM*, Mar. 2004.
[4] S. Shin and H. Schulzrinne, "Call Admission Control in IEEE 802.11 WLANs Using QP-CAT," *Proc. IEEE INFOCOM*, Apr. 2008.
[5] J. Yu et al., "Supporting VoIP Services in IEEE 802.11e WLANs," *Proc. Int'l ICST Conf. Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine)*, Nov. 2009.
[6] L. Cai et al., "Voice Capacity Analysis of WLAN with Unbalanced Traffic," *IEEE Trans. Vehicular Technology*, vol. 55, no. 3, pp. 752-761, May 2006.
[7] Y. Cheng et al., "A Cross-Layer Approach for WLAN Voice Capacity Planning," *IEEE J. Selected Areas in Comm.*, vol. 25, no. 4, pp. 678-688, May 2007.
[8] IEEE 802.11-2012, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications (Revision of IEEE Std. 802.11-2007)*, IEEE std., Mar. 2012.
[9] B.G. Lee and S. Choi, *Broadband Wireless Access and Local Networks: Mobile WiMAX and WiFi*. Artech House Publishers, 2008.
[10] A. Kamerman and L. Monteban, "WaveLAN-II: A High-Performance Wireless LAN for the Unlicensed Band," *Bell Labs Technical J.*, vol. 2, no. 3, pp. 118-133, Fall 1997.
[11] M. Lacage et al., "IEEE 802.11 Rate Adaptation: A Practical Approach," *Proc. Seventh ACM Int'l Symp. Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, Oct. 2004.
[12] J. Kim et al., "CARA: Collision-Aware Rate Adaptation for IEEE 802.11 WLANs," *Proc. IEEE INFOCOM*, Apr. 2006.
[13] S. Kim et al., "Collision-Aware Rate Adaptation in Multi-Rate WLANs: Design and Implementation," *Computer Networks*, vol. 54, no. 17, pp. 3011-3030, Dec. 2010.
[14] S.H.Y. Wong et al., "Robust Rate Adaptation for 802.11 Wireless Networks," *Proc. ACM MobiCom*, Sept. 2006.
[15] J. Bicket, "Bit-Rate Selection in Wireless Networks," master's thesis, MIT, 2005.
[16] Minstrel Rate Control Algorithm, http://wireless.kernel.org/en/developers/Documentation/mac80211/RateControl/minstrel, 2014.
[17] The Network Simulator - *ns*-3, http://www.nsnam.org/, 2014.
[18] MadWiFi: Multiband Atheros Driver for WiFi, http://madwifi-project.org/, 2014.
[19] *ITU-T Recommendation G.711, Pulse Code Modulation (PCM) of Voice Frequencies*, Int'l Telecommunications Union, Nov. 1988.
[20] *ITU-T Recommendation G.722, 7 kHz Audio-Coding within 64 kbit/s*, Int'l Telecommunications Union, Dec. 1998.
[21] *ITU-T Recommendation G.723, Dual Rate Speech Coder for Multimedia Communications Transmitting at 5.3 and 6.3 kbit/s*, Int'l Telecommunications Union, Mar. 1996.
[22] *ITU-T Recommendation G.729, Coding of Speech at 8 kbit/s Using Conjugate-Structure Algebraic-Code-Excited Linear Prediction (CS-ACELP)*, Int'l Telecommunications Union, Jan. 2007.
[23] *ITU-T Recommendation G.722.2, Wideband Coding of Speech at Around 16 kbit/s Using Adaptive Multi-Rate Wideband (AMR-WB)*, Int'l Telecommunications Union, Jan. 2002.
[24] L. Ding et al., "Measurement of the Effects of Temporal Clipping on Speech Quality," *IEEE Trans. Instrumentation and Measurement*, vol. 5, no. 4, pp. 1197-1203, Aug 2006.
[25] *IEEE 802.20 WG on Mobile Broadband Wireless Access, VoIP Evaluation for Mobile Broadband Wireless Access (MBWA): Rev. 1*, IEEE 802.20 WG, Sept. 2005.
[26] *ITU-T Recommendation G.107, The E-model, a Computational Model for Use in Transmission Planning*, Int'l Telecommunications Union, Dec. 1998.
[27] S. Moller et al., "Impairment Factor Framework for Wide-Band Speech Codecs," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 14, no. 6, pp. 1969-1976, Nov. 2006.
[28] AT&T Global IP Network, http://ipnetwork.bgtmo.ip.att.net/pws/index.html, 2014.
[29] IP Latency Statistics, http://www.verizonbusiness.com/about/network/latency/, 2014.
[30] M. DeVirgilio et al., "Internet Delay Statistics: Measuring Internet Feel Using a Dichotomous Hurst Parameter," *Proc. IEEE Southeastcon*, Mar. 2012.
[31] M. Vutukuru et al., "Cross-Layer Wireless Bit Rate Adaptation," *Proc. ACM SIGCOMM*, Aug. 2009.
[32] S. Sen et al., "Accurate: Constellation Based Rate Estimation in Wireless Networks," *Proc. Seventh USENIX Conf. Networked Systems Design and Implementation (USENIX NSDI)*, Apr. 2010.
[33] D. Halperin et al., "Predictable 802.11 Packet Delivery from Wireless Channel Measurements," *Proc. ACM SIGCOMM*, Aug. 2010.
[34] E. Perahia and R. Stacey, *Next Generation Wireless LANs: Throughput, Robustness, and Reliability in 802.11n*. Cambridge Univ. Press, 2008.
[35] B. Sadeghi et al., "Opportunistic Media Access for Multirate Ad Hoc Networks," *Proc. ACM MobiCom*, July 2002.
[36] T.S. Rappaport, *Wireless Communications: Principle and Practice*, second ed. Prentice-Hall, 2002.
[37] G. Holland et al., "A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks," *Proc. ACM MobiCom*, pp. 236-251, July 2001.
[38] IEEE 802.11n, *Amendment 5 to Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Enhancements for Higher Throughput*, IEEE std., Oct. 2009.
[39] S. Byeon et al., "Designing Adaptive RTS for MadWifi-Based WLAN Device," *Proc. Fourth Int'l Conf. Ubiquitous and Future Networks (ICUFN)*, July 2012.
[40] *IxChariot*, http://www.ixchariot.com/products/datasheets/ixchariot.html, 2014.
[41] AirPcap NX: USB 802.11a/b/g/n Adapter, http://www.riverbed.com, 2014.

Hyewon Lee (S'06) received the BE degree from School of Electrical Engineering from Seoul National University (SNU), Seoul, Korea in 2006. He is currently working toward the PhD degree in the School of Electrical Engineering at Seoul National University, Seoul, Korea. His research interests include cognitive radios, third generation partnership project long term evolution (3GPP-LTE), performance evaluation for wireless networks, in particular, IEEE 802.11 wireless local-area networks (WLANs). He is a student member of the IEEE.

Seongho Byeon (S'12) received the BE and MS degrees from the School of Electrical Engineering from Seoul National University (SNU), Seoul, Korea, in 2011 and 2013, respectively. He is currently working toward the PhD degree in the School of Electrical Engineering at Seoul National University, Seoul, Korea. His research interests include protocol design and performance evaluation of wireless/mobile network including IEEE 802.11 wireless local-area networks (WLANs). He is a student member of the IEEE.

Byoungjin Kim received the BS and MS degrees all from the School of Electrical Engineering, Seoul National University (SNU), Korea, in 2010 and 2012, respectively. He has been with SK Telecom, Korea as an engineer since March 2012. His research interests include IEEE 802.11 wireless local-area networks (WLANs), Voice over LTE (VoLTE), and IP multimedia subsystem (IMS).

Kwang Bok Lee (M'86-F'11) received the BASc and MEng degrees from the University of Toronto, Toronto, Canada, in 1982 and 1986, respectively, and the PhD degree from McMaster University, Canada, in 1990. He was with Motorola Canada from 1982 to 1985, and Motorola USA from 1990 to 1996 as a senior staff engineer. At Motorola, he was involved in the research and development of wireless communication systems. He was with Bell-Northern Research, Canada, from 1989 to 1990. In March 1996, he joined the School of Electrical Engineering, Seoul National University, Seoul, Korea. Currently, he is a professor in the School of Electrical Engineering. He has been the chair of the School of Electrical Engineering since 2011, and was a director of Institute of New Media and Communications from 2007 to 2009. He has been serving as a consultant to a number of wireless industries. His research interests include mobile communications and communication techniques covering physical layer and upper layer. He holds 25 US patents and 28 Korean patents, and has a number of patents pending. He was an editor of the *IEEE Journal on Selected Areas in Communications*, Wireless Series in 2001, and has been an editor of the *IEEE Transactions on Wireless Communications* since 2002. He was a division editor of the *Journal of Communications and Networks (JCN)* in 2006. He was a director of *JCN* from 2007 to 2008. Currently, he is a vice president of Korea Information and Communications Society (KICS). He was a cochair of the International Conference on Communications (ICC2005) Wireless Communication Symposium. He received the Best Paper Award from the CDMA International Conference 2000 (CIC 2000), and the Best Teacher Award in 2003 and 2006, respectively, from College of Engineering, Seoul National University. He received the Special Award from Samsung Advanced Institute of Technology in 2005. He is a fellow of the IEEE.

Sunghyun Choi (S'96-M'00-SM'05-F'14) received the BS (summa cum laude) and MS degrees from Korea Advanced Institute of Science and Technology in 1992 and 1994, respectively, and the PhD degree from The University of Michigan, Ann Arbor, in 1999. He is a professor in the Department of Electrical and Computer Engineering, Seoul National University (SNU), Korea. Before joining SNU in 2002, he was with Philips Research USA. He was also a visiting associate professor at Stanford University from June 2009 to June 2010. His current research interests are in the area of wireless/ mobile networks. He authored/coauthored more than 160 technical papers and book chapters in the areas of wireless/mobile networks and communications. He has co-authored (with B.G. Lee) a book *Broadband Wireless Access and Local Networks: Mobile WiMAX and WiFi,* Artech House, 2008. He holds over 100 patents, and has tens of patents pending. He has served as a general cochair of COMSWARE 2008, and a Technical Program Committee cochair of ACM Multimedia 2007, IEEE WoWMoM 2007, and COMSWARE 2007. He has also served on program and organization committees of numerous leading wireless and networking conferences including ACM MobiCom, IEEE INFOCOM, IEEE SECON, and IEEE WoWMoM. He is also currently serving on the editorial boards of the *IEEE Transactions on Mobile Computing* and *IEEE Wireless Communications*. He has served as a guest editor of the *IEEE Journal on Selected Areas in Communications (JSAC)* and *ACM Wireless Networks (WINET)*. From 2000 to 2007, he was an active voting member of IEEE 802.11 WLAN Working Group. He has received a number of awards including the Presidential Young Scientist Award (2008); IEEK/IEEE Joint Award for Young IT Engineer (2007); Shinyang Scholarship Award (2011); the Outstanding Research Award (2008) and the Best Teaching Award (2006) from the College of Engineering, SNU; and the Best Paper Award from IEEE WoWMoM 2008. He is a fellow of the IEEE, and a member of the ACM, KICS, IEEK, KIISE.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.