An Efficient Rate-Adaptive MAC for IEEE 802.11

Yuanzhu Peter Chen[1], Jian Zhang[2], and Anne N. Ngugi[1]

[1] Department of Computer Science, Memorial University of Newfoundland, Canada
[2] CAIP, Rutgers University, USA

Abstract. Data rate selection for IEEE 802.11-based wireless networks is not specified in the Specification. The problem of determining an appropriate data rate for the sender to send DATA frames and to adapt to changing channel conditions is referred to as *rate adaptation*. We propose DRA (differential rate adaptation), a rate adaptation scheme for IEEE 802.11 networks. It enables a high network throughput by adaptively tuning the data transmission rate according to the channel conditions. It is responsive to link quality changes and has little implementation overhead. Our experiments indicate that DRA yields a throughput improvement of about 20% to 25% compared to previous work.

1 Introduction

Mobile communication is becoming an integral component of a new era of life style. Along with other forms of wireless networks, mobile ad hoc and mesh networks provide a flexible yet economical platform for short- and mid-range communications. The most prevalent technology to implement these networks is the IEEE 802.11 compliant devices. The PHY layer of the IEEE 802.11 standard family provides a set of different modulation schemes and data rates for use in different channel conditions. The standard itself, however, does not specify how these data rates are selected adaptively. Therefore, the problem of determining an appropriate modulation scheme and thus a reasonable data rate attracts interests in the research of wireless networking.

Existing rate adaptation schemes in the literature generally fall into two categories. At one extreme, there are "open-loop" solutions that allows the sender to continually probe higher data rate. The ARF of Lucent WaveLAN-II [1] is an example. At the other extreme, there are "closed-loop" solutions where a sender explicitly solicits the receiver to provide reception quality information to determine an appropriate data rate. Examples of this latter approach include RBAR [2] and OAR [3]. To avoid mistaken rate reduction at the sending side and, thus, to further improve the performance of the closed-loop rate-adaptation schemes as above, a sender should differentiate the causes of lost DATA frames to reach more informed decisions, such as in LD-ARF [4] and CARA [5]. The closed-loop approaches intend to solve the *blind probing* of the open-loop approaches, where a sender keeps trying sending a DATA frame at a higher data rate from time

H. Zhang et al. (Eds.): MSN 2007, LNCS 4864, pp. 233–243, 2007.

to time, even though the receiver can not actually handle a faster transmission. The result of such blindness is the loss of frames transmitted at overly high data rates. The cost to overcome such a blindness problem is the mandatory use of the RTS/CTS control frames to measure the channel condition at the receiving side. This introduces extra overhead since RTS/CTS frames can be disabled optionally in the original IEEE 802.11 DCF for higher network throughput.

In this work, we propose a rate adaptation scheme that combines the advantages of the open-loop and the closed-loop approaches, called Differential Rate Adaptation (DRA). In particular, we use a single RTS/CTS exchange between a given sender-receiver pair to lead multiple DATA/ACK dialogs in the sequel. Each ACK contains in its header a bit to indicate the sender if the next higher data rate is recommended or not according to the reception of the previous DATA frame. Use of this feedback to the sender also provides a precision tolerance of the earlier channel quality estimation via RTS/CTS. Such a design follows a similar rationale of the Explicit Congestion Notification (ECN) as the TCP/IP architecture. The benefit of doing so is to avoid undesired outcomes before they happen rather than recovering from bad situations after they have occurred. In case of a lost DATA frame, the retransmit may be done at a lower rate.

2 Related Work

In wireless communications, rate adaptation is a mechanism for the sender to determine an appropriate data transfer rate to use the channel to the maximum extent. Due to the transient nature of channel conditions, such a mechanism must be responsive to the changes with a small overhead. Here, we focus on rate adaptation mechanisms proposed for the IEEE 802.11 based wireless networks, infrastructured or ad hoc. Since rate adaptation is not part of the IEEE 802.11 Specifications [6], the design of these mechanisms varies considerably. Depending on the scope of information that a sender uses to make the decision on rate selection, these mechanisms are usually divided into two categories, open loop and closed loop. In an open loop approach, the sender makes the decision solely based on its own perception, such as the outcome of a previous DATA transmission or the reception quality of an ACK. In a closed loop design, the sender explicitly solicits the receiver to estimate the channel condition and to feed this information back to the sender to select an appropriate data rate. In this section, we review some typical proposals of rate adaptation, open loop followed by closed loop.

The first and most widely adopted open loop rate adaptation in 802.11 devices is the ARF (auto-rate fallback) of WaveLAN-II of Lucent Technologies [1]. It consists rate probing and fallback, an idea similar to various TCP congestion control protocols. After a certain number (10 by default) of consecutive successful DATA transmissions at a given data rate, and if there is a higher data rate available, the sender selects a higher rate for the subsequent transmissions. If the channel can sustain the higher rate for a number of DATA frames, the next higher data rate is probed. If, however, a DATA transmission fails (one retrial is allowed for each data rate by default), the sender falls back to a lower data rate to retransmit the same DATA frame. In this case, a further fallback will be needed if the retrial of the transmission fails, too. ARF is simple and works fairly well. Some variants of ARF have been proposed, e.g. the FER (frame error rate) based approach [7]. In addition to using purely link level observations such as the outcomes of DATA frames, some other open loop proposals go further to use information provided by the PHY layer, such as SINR (signal to interference and noise ratio) or RSS (received signal strength) [8,9]. An important but reasonable assumption of these protocols is the symmetry of channel conditions. As a result, the sender can look up a good data rate from a pre-established table based on the SINR or RSS provided by its own PHY layer, hoping that the receiver is experiencing something similar.

Thus far, the open loop mechanisms implicitly assume that the loss of a DATA frame is caused by bad channel conditions and can be relieved by reducing the data rate. However, in a dense, especially multi-hop, wireless network, this can be well caused by collisions. Reducing data rate regardless of its actual causes not only brings down the network throughput but can also cause further collisions due to a longer transmission time of the same DATA frame. Observing this, LD-ARF (loss-differentiating ARF) [4] and CARA (collision-ware rate adaptation) [5] are proposed as smart rate fallback mechanisms by differentiating the causes of a lost DATA frame. Both LD-ARF and CARA probe for higher data rate as earlier open loop proposals. But when losing a DATA frame, the sender falls back to a lower rate only if it believes that the DATA loss was caused by a bad channel; otherwise, it simply retransmits the frame at the same data rate. LD-ARF and CARA differ in the way that they deduce the causes of a lost DATA frame. In LD-ARF, two loss differentiation methods are used, depending on whether RTS/CTS is used. In the RTS/CTS mode, the loss of a DATA frame after a successful reception of a CTS frame is considered to be caused by bad channel conditions. This is because of 802.11's robustness in the transmitting control frames, both in terms of modulation and duration. In this case, a lower data rate should be used. On the other hand, if an expected CTS is missing, the DATA frame should be transmitted at the same data rate because of the collision signified by the lost RTS. In the basic mode where RTS/CTS is disabled, and also assuming that there are no hidden terminals, a garbled DATA frame trigger the receiver to transmit a NAK (negative acknowledgment) if the MAC header of the frame can be reconstructed correctly. The rationale for LD-ARF is that, even when the channel condition is so bad that the entire frame is garbled, the MAC header can still be intact because of its small length, given a fixed BER. Thus, a NAK signifies the sender of bad channel conditions while losing a DATA frame without a NAK coming back indicates a collision. CARA also has two methods to detect collisions. The first one is similar to that of LD-ARF. That is, a successful RTS/CTS exchange followed by a lost DATA frame indicates a bad channel condition. Realizing the communications overhead of enabling RTS/CTS, CARA employs an RTS activation mechanism. The RTS/CTS are used only occasionally for diagnostic purposes.

The overhead of the open loop design is small in terms of extra bandwidth consumed. Another advantage is that it does not require the modification of the frame forms defined by the Specifications. On the flip side, the information scope used by any open loop approach is limited. A sender draws a decision upon its own perception, may it be from the link layer or physical layer. In addition, the constant attempt to transmit at a higher data rate can affect the network throughput negatively. Loss differentiation may improve the performance to a degree by avoiding reducing data rate mistakenly, but the deduction of frame collision is not sufficiently accurate, especially when there are hidden or masked nodes. Closed loop approaches attempt to make more informed decisions with the help of the receiver. Indeed, whether a DATA frame can be received correctly at a given data rate can only be estimated much more precisely on the receiving side. The cost of transferring the information from the receiving side is an increased overhead in protocol implementation.

RBAR (receiver-based auto rate) [2] is the first closed loop protocol in the context of IEEE 802.11 networks. In RBAR, a sender always transmits an RTS frame before transmitting a DATA frame. Upon receiving the RTS, the receiver also measures the SINR of the moment. Based on acceptable BER, the receiver looks up the highest data rate that the transient SINR supports. This data rate is fed back to the sender using a modified CTS frame. OAR (opportunistic auto-rate) [3] enhances RBAR using fragmentation of the IEEE 802.11 MAC. It improves the efficiency of RBAR significantly by allowing a single RTS/CTS exchange to lead a train of DATA/ACK pairs. This overcomes the major disadvantage of low efficiency of the closed loop design. This idea is further extended by MAD (medium access diversity) [10]. MAD is designed to improve network throughput by allowing a sender to choose a neighbor that can receive a DATA frame at the highest data rate. In essence, MAD uses a link level anycast mechanism, where the RTS format is extended to include a list of multiple receiver addresses. Such an extension is also made to solve the HOL (head of line) blocking problem in mesh networks, referred to as MRTS (multicast RTS) [11].

Using additional information from the receiving side, the closed loop design usually is more responsive to channel condition changes. Its overhead can be reduced by allowing a data burst after a single RTS/CTS dialog. Still, it is susceptible to inaccurate channel estimation and the length of the data burst is heavily constrained by the channel coherence time. DRA (differential rate adaptation) proposed in this work combines the advantages of both open and closed loop approaches to achieve better performance, as described in the next section.

3 Design of DRA

DRA uses a single RTS/CTS dialog to lead a burst of DATA/ACK pairs. This is essentially a combination of the closed and open loop designs. The probing of a higher data rate is done with an effective differential compensation, i.e. using a flag in the ACK header, without the risk of using too high a data rate for the channel to sustain. This is done without extra overhead and with maximum

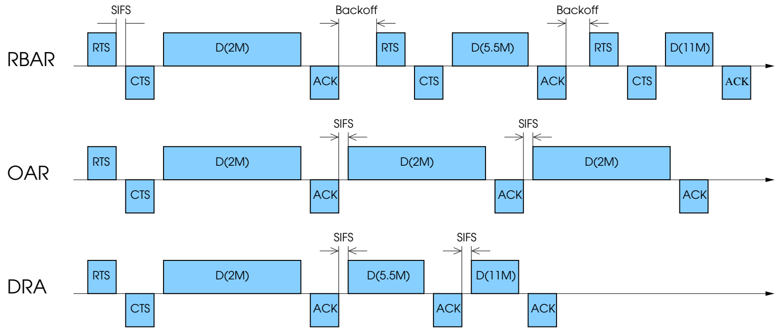

Fig. 1. Design of DRA

compatibility with the Specifications. As a result, during the entire data burst, our single-bit feedback mechanism compensates the moderate channel condition changes, while more significant changes will be captured by the RTS/CTS leading each data burst. Another advantage of DRA is its tolerance in the inaccuracy in channel conditions estimated by the RTS/CTS exchange; the data rate adopted by the sender matches the channel quality better and better as the burst goes on. A schematic comparison between DRA with RBAR and OAR is illustrated in Fig. 1.

3.1 Data Rate Estimation and Feedback — Receiving Side Story

DRA enlists the receiver to feed the channel condition information to the sender to close its control loop. To do that, when the receiver receives the RTS or DATA frame, it also records the SINR when the frame was received. Based on that, the receiver can look up from a table the highest data rate that the recorded SINR supports with an acceptable bit error rate. Then the receiver puts such a planned data rate in the CTS frame so that the sender can adopt this rate in the subsequent burst of DATA frames. Further, the estimation errors and the channel condition changes can be compensated by piggy-backing a single bit in the ACK from the receiver to indicate if the next higher data is feasible for the next DATA frame in the burst.

Once the receiver has determined the data rate r_i, it needs to feed this information back to the sender. To do that, we change the definition of the "duration" field of a MAC header, as in RBAR and OAR. In the Specification, the duration field is a standard 16-bit field of a data or control frame. The value is the amount of time needed before the subsequent ACK is received in milliseconds. This is used to set the NAV (network allocation vector) of a node that overhears the frame to accomplish VCS (virtual carrier sensing). Here, it is changed to two subfields, *rate* and *length*, of 4 and 12 bits, respectively (Fig. 2). The rate subfield is sufficient to address 16 different data rates, which is sufficient to represent the data rates required by 802.11 Legacy, 11b, and 11a/g. (That is, 1M and 2M from Legacy, 5.5M and 11M from 11b, and 6M, 9M, 12M, 18M, 24M, 36M, 48M, and

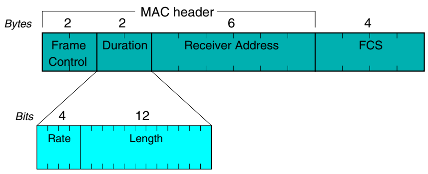

Fig. 2. CTS frame format of DRA

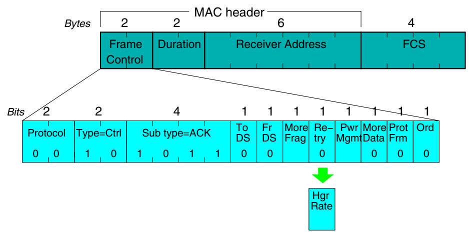

Fig. 3. ACK frame format of DRA

54M from 11a/g) The length subfield contains the length of the MSDU (frame body) in bytes. The 12 bits therein can potentially represent an MSDU of 4096 bytes, which is greater than the maximum body size of 2346 bytes. Using these two subfield, a node that overhears the header can reconstruct the NAV value for VCS. When constructing the CTS frame, the receiver puts the data rate index i in the rate subfield and copies the value of the length subfield in the incoming RTS frame header. It then transmits the CTS frame to the sender at the basic data rate.

Similarly, the receiver also indicates to the sender if a higher data rate should be adopted using a single bit. To do that, we utilize the "retry" bit in the *frame control* field of the MAC header since it is redundant for the ACK frame (Fig. 3). We call such a bit the "higher rate" flag. When the receiver receives a DATA frame, it also estimates the highest data rate r_i that could have been used by that frame transmission. If the channel condition has improved significantly so that r_i is higher than the data rate at which the DATA was received, the higher rate flag is set to 1 to inform the sender that the next higher data rate can be used for the subsequent DATA frame. Otherwise, the flag is set to 0.

3.2 Adaptive DATA Burst—Sending Side Story

In DRA, a sender contends for the channel before exchanging RTS/CTS with the receiver. Then, a burst of DATA/ACK pairs will be transmitted between the sending and receiving parties. The inter-frame space between each of these consecutive frames is SIFS (as in Fig. 1), so that the train of frames will not be interrupted by other nodes. This burst of DATA/ACK frames is responsible for adapting to channel condition changes and for retransmitting garbled packets.

The data rate $R = r_i$ $(1 \leq i \leq k)$ of the first DATA frame is determined by the value set in the "rate" field of the received CTS frame. The sender then constructs the DATA frame and transmits the frame at R Mbps. It then waits for the ACK to indicate if the transmission was successful. If so, it will transmit the next DATA frame; otherwise, it must retransmit the same DATA frame. In the first case, whether a higher data rate should be used for the next DATA frame is determined by the "higher rate" flag in the ACK header. If the flag is set to 1, it sets R to $r_{\min\{k,i+1\}}$, i.e. to the next available higher data rate, to explore for higher throughput. If the flag is 0, it remains at the same data rate

r_i. In the second case, where the expected ACK was missing, the same DATA frame is retransmitted at the same data rate.

In the design of DRA, the temporal length of the burst can be considerably longer than that needed to complete the RTS/CTS/DATA/ACK 4-way handshake at the basic rate due to its adaptiveness to channel conditions. We denote this burst length by T_b. That is, as long as the queue within the sender is non-empty, it keeps transmitting the next DATA frame after SIFS of receiving the ACK. The sender keeps pumping data through the wireless channel until T_b seconds has elapsed since the moment it started sending the first DATA frame of the burst. The choice of T_b should satisfy that, during this amount of time, the channel condition at the receiver can be compensated by the rate adaptation mechanism as described above. In our implementation, T_b is set to 50 ms, which is approximately the time to transmit slightly over two DATA frames of maximum size (2346 bytes) at the basic data rate of 1 Mbps. Such a choice of T_b is verified by the calculation in OAR. In contrast, T_b is set to 20ms in OAR to accommodate one where the coherence time is about 122.88 (24.57, 12.28, 6.14, resp.) ms for a center frequency of 2.4 GHz at mobile speed of 1 (5, 10, 20, resp.) m/s. After completing the burst, the sender must contend for the channel, as specified in the DCF (distributed coordination function) of the Specifications [6], if it has more data to transmit.

3.3 Setting the NAV — Everybody Else

An 802.11 device can be used to implement a multi-hop wireless network. To cope with the hidden terminal problem, the duration information is embedded in each type of frame, so that, whenever a node overhears the frame, it stays away from the channel for the indicated amount of time. That is, the network allocation vector is set to the duration value. As discussed earlier, DRA differs from the Specifications in that its duration field has two components (Fig. 2), the rate index i and the payload size S. Note that the PHY layer header and MAC headers of a frame are all transmitted at the fixed basic rate. Thus, the only variables that contribute the the transmission time of a data frame are the data rate r_i and the size of payload S. We further denote the time needed to transmit the PHY (RTS, CTS, DATA, and ACK, resp.) header by H_p (H_r, H_c, H_d, and H_a, resp.). When a node overhears a frame, it sets if NAV as follows:

- RTS — $SIFS + H_p + H_c$. That is, when overhearing an RTS, the NAV should be set to secure the channel until point A in Fig. 4.
- CTS — $SIFS + H_p + H_d + 8 \times S/r_i$. That is, when overhearing a CTS, the NAV should be set to secure the channel until point B.
- DATA — $SIFS + H_p + H_a$. That is, when overhearing a DATA, the NAV should be set to secure the channel until point C.
- ACK — $SIFS + H_p + H_d + 8 \times S/r_i$. That is, when overhearing an ACK, the NAV should be set to secure the channel until point D.

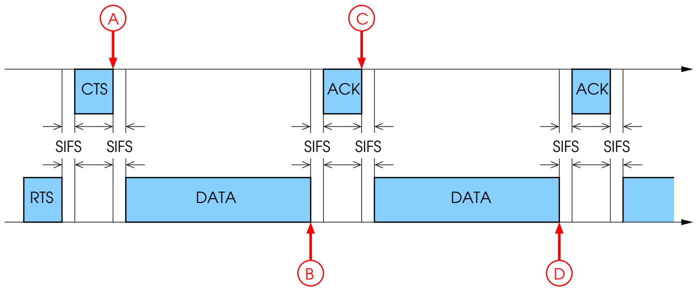

Fig. 4. Setting the NAV

4 Simulation

To study the effectiveness of DRA, we resort to packet level simulation using ns-2. The focus is to study the data link layer throughput of DRA in highly dynamic channel conditions with and without hidden terminals.

We use the pre-computed time series data of Punnroose et al. [12] to simulate a rapidly fading channel that follows the Ricean distribution. In the simulation, we vary K, the Ricean parameter, between 0 and 5 to achieve different levels of contribution of the line-of-sight component in the received signal. Since DRA is an extension of OAR, we compare these two protocols' performance in the same changing channel condition. Our preliminary experiments showed that with relatively low node mobility, say speed of 2.5m/s (setting the maximum Doppler frequency f_m to 30Hz), DRA offers a slightly higher throughput. This also indicates that OAR's succinct design is fairly effective for a low to medium mobility rate. In contrast, DRA's per-fragment rate adaptation achieves higher throughput even if the channel conditions change rapidly. This is verified by our experiment below when setting f_m to 300Hz (i.e. 25m/s of maximum mobility velocity). Furthermore, DRA enables long fragment bursts to reduce the control overhead, thus, having higher efficiency.

The testing is done in two scenarios, without and with hidden nodes (Fig. 5). In each scenario, we set the fragment burst length to 6ms and 50ms, respectively, to find out about the effect of using a longer burst length. Apparently, longer bursts reduce the protocol overhead introduced by the RTS/CTS handshake at the beginning of each burst. However, the changing channel condition will render

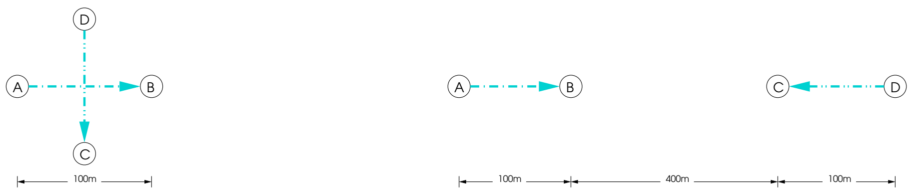

Fig. 5. Simulation scenarios

that the data rate estimated by the receiver and fed back via CTS to be invalid as the burst goes on. Without a compensation mechanism, the burst length can be rather restricted in a rapidly dynamic environment. In our simulation, we observe how DRA benefits from its adaptiveness to support longer bursts well.

In the first scenario (Fig. 5, left), we deploy two CBR flows (A to B and D to C), each of which can saturate the network capacity. The transmitter-receiver separation distance is 100m, such that the data rate fluctuates between 1 and 11Mbps. Since all nodes in this scenario are not farther than 100m apart, they can, in most cases, decode the control frames (i.e. RTS, CTS, and ACK) and the header of DATA frames. Therefore, these nodes are fully connected to each other. For both OAR and DRA, we set the burst length to 6ms and 50ms. We start the two flows at the beginning of the simulation simultaneously. The simulation has a duration of 50 seconds and is repeated 10 times. We measure the number of packets aggregated for the two flows per unit of time. We observed that the measurement stabilized in a short time. For a fixed Ricean parameter K, we plot the total throughput, i.e. number of packet received in 50 seconds, for each protocol-burstlength combination (DRA vs OAR and 6ms vs 50ms) as depicted in Fig. 6. In the plot, we see that the throughput increases as the line-of-sight component becomes stronger (larger K) for each combination. In addition, for the shorter burst length of 6ms, DRA possesses an average of about 4% of throughput gain over OAR. However, for the longer burst length of 50ms, DRA's adaptiveness introduces an average of about 25% of throughput gain.

In the second scenario (Fig. 5, right), we also deploy two CBR flows (A to B and D to C), But here, we separate these flows fairly far away such that the two receivers (B and C) are 400m apart and the senders (A and D) are 600m apart. As a result of the ns-2 default settings, which is fairly typical in this aspect in reality, the two pairs are hidden from each other but the two receivers are still within the carrier sensing range of both senders. Ideally in this case, the two flows should be transported in parallel. But due to the fact that the NAV cannot be set effectively by a distant transmitter, there will not be a 100% parallelism. Simulation done for this scenario is plotted in Fig. 7, and it

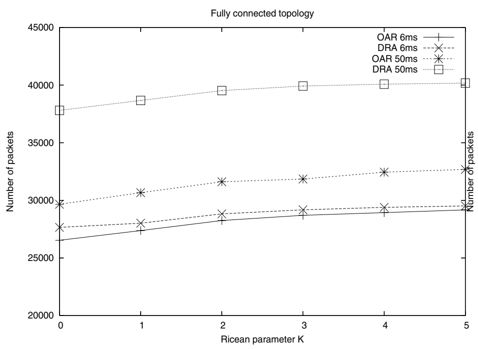

Fig. 6. Full connection

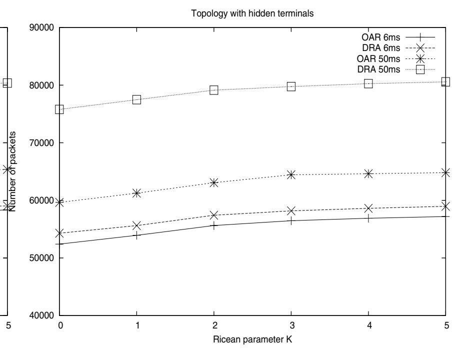

Fig. 7. With hidden nodes

indicates that the effect of hidden transmitters is minimized and the throughput is roughly doubled in the matching point in the previous scenario. Here, the 6ms burst length enables DRA an approximately 3% of throughput gain and the longer 50ms burst length offers a more significant 20% throughput gain.

5 Conclusion and Future Work

In this work, we investigate a data rate selection method for IEEE 802.11 devices. In particular, we present a feedback mechanism, DRA, for a receiver to compensate channel condition changes. This is essentially combining the advantages of the open- and closed-loop designs for the rate adaptation solutions in the literature. The design of DRA has zero extra overhead compared to its ancestor, OAR. In our simulation, DRA indicates a 20% to 25% throughput gain when using a longer burst length. In the research to follow, we plan to integrate loss differentiation to avoid blind rate fallback caused by a lost fragment. Our preliminary work indicated that a simple inclusion of the CCA-based or the transmission history based approaches in the literature does not offer a noticeable improvement of DRA. Nevertheless, more sophisticated decision mechanisms may further improve DRA.

References

1. Kamerman, A., Monteban, L.: WaveLAN-II: a high-performance wireless LAN for the unlicensed band. Bell Labs Technical Journal 2(3), 118–133 (1997)
2. Holland, G., Vaidya, N., Bahl, P.: A rate-adaptive MAC protocol for multi-hop wireless networks. In: Proceedings of MobiCom, Rome, Italy, pp. 236–251 (July 2001)
3. Sadeghi, B., Kanodia, V., Sabharwal, A., Knightly, E.: Opportunistic media access for multirate ad hoc networks. In: Proceedings of MobiCom, Atlanta, GA, pp. 24–35 (September 2002)
4. Pang, Q., Leung, V.C.M., Liew, S.C.: A rate adaptation algorithm for IEEE 802.11 WLANs based on MAC-layer loss differentiation. In: BroadNets. Proceedings of the Seconnd International Conference on Broadband Networks, Boston, MA, pp. 659–667 (October 2005)
5. Kim, J., Kim, S., Choi, S., Qiao, D.: CARA: Collision-aware rate adaptation for IEEE 802.11 WLANs. In: Proceedings of INFOCOM, Barcelona, Spain, pp. 146–157 (April 2006)
6. Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications – 802.11. Institute of Electronic and Electrical Engineers (IEEE) (1999)
7. Braswell, B., McEachen, J.: Modeling data rate agility in the IEEE 802.11a WLAN protocol. In: Proceedings of OPNETWORK (2001)
8. Qiao, D., Choi, S., Shin, K.G.: Goodput analysis and link adaptation for IEEE 802.11a wireless LANs. IEEE Transactions on Mobile Computing 1(4), 278–292 (2002)

9. del Prado Pavon, J., Choi, S.: Link adaptation strategy for IEEE 802.11 WLAN via received signal strength measurement. In: Proceedings of ICC, pp. 1108–1113 (May 2003)
10. Ji, Z., Yang, Y., Zhou, J., Takai, M., Bagrodia, R.: Exploiting medium access diversity in rate adaptive wirless LANs. In: Proceedings of MobiCom, Philadelphia, PA, pp. 345–359 (September 2004)
11. Zhang, J., Chen, Y.P., Marsic, I.: Adaptive MAC scheduling using channel state diversity for wireless networks. In: Proceedings of WiCom (September 2006)
12. Punnroose, R.J., Nikitin, P.N., Stancil, D.D.: Efficient simulation of Ricean fading within a packet simulator. In: Proceedings of VTC, pp. 764–767 (September 2000)